Joaquim Filipe
José Cordeiro (Eds.)

Enterprise Information Systems

LNBIP 24

11th International Conference, ICEIS 2009
Milan, Italy, May 2009
Proceedings

Springer

Lecture Notes
in Business Information Processing 24

Series Editors

Wil van der Aalst
Eindhoven Technical University, The Netherlands
John Mylopoulos
University of Trento, Italy
Norman M. Sadeh
Carnegie Mellon University, Pittsburgh, PA, USA
Michael J. Shaw
University of Illinois, Urbana-Champaign, IL, USA
Clemens Szyperski
Microsoft Research, Redmond, WA, USA

Joaquim Filipe José Cordeiro (Eds.)

Enterprise Information Systems

11th International Conference, ICEIS 2009
Milan, Italy, May 6-10, 2009
Proceedings

Springer

Volume Editors

Joaquim Filipe
José Cordeiro
Institute for Systems and Technologies of Information
Control and Communication (INSTICC) and
Instituto Politécnico de Setúbal (IPS)
Department of Systems and Informatics
Rua do Vale de Chaves, Estefanilha, 2910-761 Setúbal, Portugal
E-mail: {j.filipe,jcordeir}@est.ips.pt

Library of Congress Control Number: Applied for

ACM Computing Classification (1998): J.1, H.3.5, H.5, I.2.11

ISSN 1865-1348
ISBN-10 3-642-01346-5 Springer Berlin Heidelberg New York
ISBN-13 978-3-642-01346-1 Springer Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer. Violations are liable to prosecution under the German Copyright Law.

springer.com

© Springer-Verlag Berlin Heidelberg 2009
Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 12664511 06/3180 5 4 3 2 1 0

Preface

This book contains the collection of full papers accepted at the 11th International Conference on Enterprise Information Systems (ICEIS 2009), organized by the Institute for Systems and Technologies of Information Control and Communication (INSTICC) in cooperation with the Association for Advancement of Artificial Intelligence (AAAI) and ACM SIGMIS (SIG on Management Information Systems), and technically co-sponsored by the Japanese IEICE SWIM (SIG on Software Interprise Modeling) and the Workflow Management Coalition (WfMC).

ICEIS 2009 was held in Milan, Italy. This conference has grown to become a major point of contact between research scientists, engineers and practitioners in the area of business applications of information systems. This year, five simultaneous tracks were held, covering different aspects related to enterprise computing, including: "Databases and Information Systems Integration," "Artificial Intelligence and Decision Support Systems," "Information Systems Analysis and Specification," "Software Agents and Internet Computing" and "Human–Computer Interaction". All tracks describe research work that is often oriented toward real-world applications and highlight the benefits of information systems and technology for industry and services, thus making a bridge between academia and enterprise.

ICEIS 2009 received 644 paper submissions from 70 countries in all continents; 81 papers were published and presented as full papers, i.e., completed research work (8 pages/30-minute oral presentation). Additional papers accepted at ICEIS, including short papers and posters, were published in the regular conference proceedings.

These aforementioned numbers, leading to a "full-paper" acceptance ratio below 13%, show the intention of preserving a high-quality forum for the next editions of this conference. Additionally, as usual in the ICEIS conference series, a number of invited talks, presented by internationally recognized specialists in different areas, contributed positively to reinforcing the overall quality of the conference and to providing a deeper understanding of the enterprise information systems field.

We hope that you find the papers included in this book interesting and we trust they may represent a helpful reference in the future for all those who need to address any of the research areas mentioned above.

March 2009

Joaquim Filipe
José Cordeiro

Organization

Conference Chair

Joaquim Filipe	Polytechnic Institute of Setúbal / INSTICC, Portugal

Program Chair

José Cordeiro	Polytechnic Institute of Setúbal / INSTICC, Portugal

Organizing Committee

Sérgio Brissos	INSTICC, Portugal
Marina Carvalho	INSTICC, Portugal
Helder Coelhas	INSTICC, Portugal
Vera Coelho	INSTICC, Portugal
Andreia Costa	INSTICC, Portugal
Bruno Encarnação	INSTICC, Portugal
Bárbara Lima	INSTICC, Portugal
Raquel Martins	INSTICC, Portugal
Carla Mota	INSTICC, Portugal
Vitor Pedrosa	INSTICC, Portugal
Vera Rosário	INSTICC, Portugal
José Varela	INSTICC, Portugal

Senior Program Committee

Senén Barro, Spain
Jean Bézivin, France
Enrique Bonsón, Spain
Albert Cheng, USA
Bernard Coulette, France
Andrea De Lucia, Italy
Jan Dietz, The Netherlands
Virginia Dignum, The Netherlands
Schahram Dustdar, Austria
António Figueiredo, Portugal
Nuno Guimarães, Portugal
Dimitris Karagiannis, Austria
Michel Leonard, Switzerland
Kecheng Liu, UK
Pericles Loucopoulos, UK
Kalle Lyytinen, USA
Yannis Manolopoulos, Greece
José Legatheaux Martins, Portugal

Masao Johannes Matsumoto, Japan
Marcin Paprzycki, Poland
Alain Pirotte, Belgium
Klaus Pohl, Germany
Matthias Rauterberg, The Netherlands
Colette Rolland, France
Narcyz Roztocki, USA
Abdel-Badeeh Salem, Egypt
Bernadette Sharp, UK
Timothy K. Shih, Taiwan
Alexander Smirnov, Russian Federation
Ronald Stamper, UK
Antonio Vallecillo, Spain
François Vernadat, France
Frank Wang, UK
Merrill Warkentin, USA

Program Committee

Lena Aggestam, Sweden
Patrick Albers, France
Vasco Amaral, Portugal
Yacine Amirat, France
Andreas Andreou, Cyprus
Colin Anthony, UK
Gustavo Arroyo-Figueroa, Mexico
Wudhichai Assawinchaichote, Thailand
Juan Carlos Augusto, UK
Anjali Awasthi, Canada
Cecilia Baranauskas, Brazil
Steve Barker, UK
Balbir Barn, UK
Daniela Barreiro Claro, Brazil
Nick Bassiliades, Greece
Nadia Bellalem, France
Orlando Belo, Portugal
Hatem Ben Sta, Tunisia
Sadok Ben Yahia, Tunisia
Manuel F. Bertoa, Spain
Minal Bhise, India
Oliver Bittel, Germany
Luis Borges Gouveia, Portugal
Danielle Boulanger, France
Jean-Louis Boulanger, France
José Ângelo Braga de Vasconcelos, Portugal
Stéphane Bressan, Singapore
Miguel Calejo, Portugal
Coral Calero, Spain
Olivier Camp, France
Gerardo Canfora, Italy
Angélica Caro, Chile
Nunzio Casalino, Italy
Maria Filomena C. de Castro Lopes, Portugal
Maiga Chang, Canada
Laurent Chapelier, France
Cindy Chen, USA
Jinjun Chen, Australia
Francesco Colace, Italy
Cesar Collazos, Colombia
Jose Eduardo Corcoles, Spain
Antonio Corral, Spain
Sharon Cox, UK
Alfredo Cuzzocrea, Italy
Jacob Cybulski, Australia
Mohamed Dahchour, Morocco
Sergio de Cesare, UK
Nuno De Magalhães Ribeiro, Portugal
José Neuman De Souza, Brazil
Suash Deb, India
Vincenzo Deufemia, Italy
Rajiv Dharaskar, India
Massimiliano Di Penta, Italy
Kamil Dimililer, Turkey

José Javier Dolado, Spain
António Dourado Correia, Portugal
Juan C. Dueñas, Spain
Barry Eaglestone, UK
Hans-Dieter Ehrich, Germany
Jean-Max Estay, France
Yaniv Eytani, USA
João Faria, Portugal
Antonio Fariña, Spain
Antonio Fernández-Caballero, Spain
Edilson Ferneda, Brazil
Paulo Ferreira, Portugal
Filomena Ferrucci, Italy
Mariagrazia Fugini, Italy
Jose A. Gallud, Spain
Juan Garbajosa, Spain
Leonardo Garrido, Mexico
Peter Geczy, Japan
Joseph Giampapa, USA
Paolo Giorgini, Italy
Raúl Giráldez, Spain
Pascual Gonzalez, Spain
Gustavo Gonzalez-Sanchez, Spain
Robert Goodwin, Australia
Jaap Gordijn, The Netherlands
Silvia Gordillo, Argentina
Feliz Gouveia, Portugal
Janis Grabis, Latvia
Sven Groppe, Germany
Rune Gustavsson, Sweden
Sissel Guttormsen Schär, Switzerland
Maki K. Habib, Japan
Lamia Hadrich Belguith, Tunisia
Abdelwahab Hamou-lhadj, Canada
Christian Heinlein, Germany
Ajantha Herath, USA
Suvineetha Herath, USA
Francisco Herrera, Spain
Peter Higgins, Australia
Wladyslaw Homenda, Poland
Wei-Chiang Hong, Taiwan
Jiankun Hu, Australia
François Jacquenet, France
Ivan Jelinek, Czech Republic
Luis Jiménez Linares, Spain
Paul Johannesson, Sweden
Michail Kalogiannakis, France
Nikos Karacapilidis, Greece
Nikitas Karanikolas, Greece
Stamatis Karnouskos, Germany
Hiroyuki Kawano, Japan
Seungjoo Kim, Republic of Korea
Marite Kirikova, Latvia
Alexander Knapp, Germany
John Krogstie, Norway
Stan Kurkovsky, USA
Rob Kusters, The Netherlands
Alain Leger, France
Kauko Leiviskä, Finland
Daniel Lemire, Canada
Carlos León De Mora, Spain
Joerg Leukel, Germany
Hareton Leung, China
Qianhui LIANG, Singapore
Therese Libourel, France
Panos Linos, USA
João Correia Lopes, Portugal
Víctor López-jaquero, Spain
Miguel R. Luaces, Spain
Christof Lutteroth, New Zealand
Mark Lycett, UK
Cristiano Maciel, Brazil
Edmundo Madeira, Brazil
Nuno Mamede, Portugal
Pierre Maret, France
Herve Martin, France
Miguel Angel Martinez Aguilar, Spain
David Martins De Matos, Portugal

Katsuhisa Maruyama, Japan
Hamid Mcheick, Canada
Engelbert Mephu Nguifo, France
Subhas Misra, USA
Michele Missikoff, Italy
Ghodrat Moghadampour, Finland
Pascal Molli, France
Francisco Montero, Spain
Paula Morais, Portugal
Fernando Moreira, Portugal
Nathalie Moreno, Spain
Haralambos Mouratidis, UK
Pietro Murano, UK
Tomoharu Nakashima, Japan
Paolo Napoletano, Italy
Rabia Nessah, France
Ana Neves, Portugal
Patrick O'Neil, USA
Hichem Omrani, Luxembourg
Peter Oriogun, UK
Claus Pahl, Ireland
José R. Paramá, Spain
Eric Pardede, Australia
Rodrigo Paredes, Chile
Maria Carmen Penadés Gramaje, Spain
Gabriel Pereira Lopes, Portugal
Laurent Péridy, France
Dana Petcu, Romania
Leif Peterson, USA
Geert Poels, Belgium
José Ragot, France
Abdul Razak Rahmat, Malaysia
Jolita Ralyte, Switzerland
Srini Ramaswamy, USA
Marek Reformat, Canada
Hajo A. Reijers, The Netherlands
Ulrich Reimer, Switzerland
Marinette Revenu, France
Simon Richir, France
David Rivreau, France
Alfonso Rodriguez, Chile
Daniel Rodriguez, Spain
Pilar Rodriguez, Spain
Oscar M. Rodriguez-Elias, Mexico
Jose Raul Romero, Spain
Francisco Ruiz, Spain
Danguole Rutkauskiene, Lithuania
Ángeles S. Places, Spain
Ozgur Koray Sahingoz, Turkey
Priti Srinivas Sajja, India
Daniel Schang, France
Isabel Seruca, Portugal
Maria João Silva Costa Ferreira, Portugal
Hala Skaf-Molli, France
Pedro Soto-Acosta, Spain
Chantal Soule-Dupuy, France
Marco Spruit, The Netherlands
Martin Stanton, UK
Janis Stirna, Sweden
Renate Strazdina, Latvia
Stefan Strecker, Germany
Chun-Yi Su, Canada
Ramayah T., Malaysia
Ryszard Tadeusiewicz, Poland
Vladimir Tarasov, Sweden
Sotirios Terzis, UK
Claudine Toffolon, France
Grigorios Tsoumakas, Greece
Theodoros Tzouramanis, Greece
Athina Vakali, Greece
Michael Vassilakopoulos, Greece
Belen Vela Sanchez, Spain
Christine Verdier, France
Maria-Amparo Vila, Spain
Bing Wang, UK
Hans Weghorn, Germany

Gerhard Weiss, Austria
Graham Winstanley, UK
Wita Wojtkowski, USA
Viacheslav Wolfengagen, Russian Federation
Robert Wrembel, Poland
Mudasser Wyne, USA
Haiping Xu, USA
Lin Zongkai, China

Auxiliary Reviewers

Michael Affenzeller, Austria
Rossana Andrade, Brazil
Hércules Antônio Do Prado, Brazil
Evandro Bacarin, Brazil
Bartosz Bebel, Poland
Ismael Caballero, Spain
Jesus R. Campaña, Spain
José María Cavero Barca, Spain
Ana Cerdeira-Pena, Spain
Fabio Clarizia, Italy
Fernando William Cruz, Brazil
Guillermo de Bernardo Roca, Spain
Andrea Delgado, Uruguay
Sergio Di Martino, Italy
Fausto Fasano, Italy
Sergio Folgar Méndez, Spain
Miguel Franklin de Castro, Brazil
Anastasios Gounaris, Greece
Carmine Gravino, Italy
Tarek Hamrouni, Tunisia
Nantia Iakovidou, Greece
Ioannis Katakis, Greece
Maria Kontaki, Greece
Susana Ladra Gonzalez, Spain
Pedro Magaña, Spain
Nicolás Marín, Spain
Javier Medina, Spain
Isabelle Mirbel, France
Mª Ángeles Moraga, Spain
Thomas Natschlaeger, Austria
Matthias Nickles, UK
Germana Nobrega, Brazil
Rocco Oliveto, Italy
Gerald Oster, France
Samia Oussena, UK
Ignazio Passero, Italy
Oscar Pedreira, Spain
Michele Risi, Italy
Eduardo Rodríguez López, Spain
Maria Dolores Ruiz, Spain
Giuseppe Scanniello, Italy
Diego Seco, Spain
Boran Sekeroglu, Cyprus
Manuel Serrano, Spain
Yoshiyuki Shinkawa, Japan
Francesco Taglino, Italy
Eleftherios Tiakas, Greece
Luigi Troiano, Italy
Athanasios Tsadiras, Greece
Juan Manuel Vara Mesa, Spain
Corrado Aaron Visaggio, Italy
Fabian Wagner, Germany
Stéphane Weiss, France

Invited Speakers

Peter Geczy	AIST, Japan
Masao J. Matsumoto	Kyushu Sangyo University, Japan
Michele Missikoff	IASI-CNR, Italy
Barbara Pernici	Politecnico di Milano, Italy
Jianchang Mao	Yahoo! Labs, USA
Ernesto Damiani	University of Milan, Italy
Mike P. Papazoglou	Tilburg University, The Netherlands

Table of Contents

Part I: Databases and Information Systems Integration

MIDAS: A Middleware for Information Systems with QoS Concerns 3
Luís Fernando Orleans and Geraldo Zimbrão

Instance-Based OWL Schema Matching 14
Luiz André P. Paes Leme, Marco A. Casanova, Karin K. Breitman, and Antonio L. Furtado

The Integrative Role of IT in Product and Process Innovation: Growth and Productivity Outcomes for Manufacturing 27
Louis Raymond, Anne-Marie Croteau, and François Bergeron

Vectorizing Instance-Based Integration Processes 40
Matthias Boehm, Dirk Habich, Steffen Preissler, Wolfgang Lehner, and Uwe Wloka

Invisible Deployment of Integration Processes 53
Matthias Boehm, Dirk Habich, Wolfgang Lehner, and Uwe Wloka

Customizing Enterprise Software as a Service Applications: Back-End Extension in a Multi-tenancy Environment 66
Jürgen Müller, Jens Krüger, Sebastian Enderlein, Marco Helmich, and Alexander Zeier

Pattern-Based Refactoring of Legacy Software Systems 78
Sascha Hunold, Björn Krellner, Thomas Rauber, Thomas Reichel, and Gudula Rünger

A Natural and Multi-layered Approach to Detect Changes in Tree-Based Textual Documents 90
Angelo Di Iorio, Michele Schirinzi, Fabio Vitali, and Carlo Marchetti

CrimsonHex: A Service Oriented Repository of Specialised Learning Objects 102
José Paulo Leal and Ricardo Queirós

A Scalable Parametric-RBAC Architecture for the Propagation of a Multi-modality, Multi-resource Informatics System 114
Remo Mueller, Van Anh Tran, and Guo-Qiang Zhang

Minable Data Warehouse 125
David Morgan, Jai W. Kang, and James M. Kang

A Step Forward in Semi-automatic Metamodel Matching: Algorithms and Tool 137
José de Sousa Jr., Denivaldo Lopes, Daniela Barreiro Claro, and Zair Abdelouahab

A Study of Indexing Strategies for Hybrid Data Spaces 149
Changqing Chen, Sakti Pramanik, Qiang Zhu, and Gang Qian

Relaxing XML Preference Queries for Cooperative Retrieval 160
SungRan Cho and Wolf-Tilo Balke

DeXIN: An Extensible Framework for Distributed XQuery over Heterogeneous Data Sources 172
Muhammad Intizar Ali, Reinhard Pichler, Hong Linh Truong, and Schahram Dustdar

Dimensional Templates in Data Warehouses: Automating the Multidimensional Design of Data Warehouse Prototypes 184
Rui Oliveira, Fátima Rodrigues, Paulo Martins, and João Paulo Moura

Multiview Components for User-Aware Web Services 196
Bouchra El Asri, Adil Kenzi, Mahmoud Nassar, Abdelaziz Kriouile, and Abdelaziz Barrahmoune

Knowledge Based Query Processing in Large Scale Virtual Organizations 208
Alexandra Pomares, Claudia Roncancio, José Abásolo, and María del Pilar Villamil

Applying Recommendation Technology in OLAP Systems 220
Houssem Jerbi, Franck Ravat, Olivier Teste, and Gilles Zurfluh

Classification and Prediction of Software Cost through Fuzzy Decision Trees 234
Efi Papatheocharous and Andreas S. Andreou

s-OLAP: Approximate OLAP Query Evaluation on Very Large Data Warehouses via Dimensionality Reduction and Probabilistic Synopses . . . 248
Alfredo Cuzzocrea

Part II: Artificial Intelligence and Decision Support Systems

A Self-learning System for Object Categorization 265
Danil V. Prokhorov

A Self-tuning of Membership Functions for Medical Diagnosis 275
Nuanwan Soonthornphisaj and Pattarawadee Teawtechadecha

Insolvency Prediction of Irish Companies Using Backpropagation and Fuzzy ARTMAP Neural Networks 287
Anatoli Nachev, Seamus Hill, and Borislav Stoyanov

Frequent Subgraph-Based Approach for Classifying Vietnamese Text Documents 299
Tu Anh Hoang Nguyen and Kiem Hoang

Random Projection Ensemble Classifiers 309
Alon Schclar and Lior Rokach

Knowledge Reuse in Data Mining Projects and Its Practical Applications 317
Rodrigo Cunha, Paulo Adeodato, and Silvio Meira

Enhancing Text Clustering Performance Using Semantic Similarity 325
Walaa K. Gad and Mohamed S. Kamel

Stereo Matching Using Synchronous Hopfield Neural Network 336
Te-Hsiu Sun

Monotonic Monitoring of Discrete-Event Systems with Uncertain Temporal Observations 348
Gianfranco Lamperti and Marina Zanella

A Service Composition Framework for Decision Making under Uncertainty 363
Malak Al-Nory, Alexander Brodsky, and Hadon Nash

A Multi-criteria Resource Selection Method for Software Projects Using Fuzzy Logic 376
Daniel Antonio Callegari and Ricardo Melo Bastos

An Optimized Hybrid Kohonen Neural Network for Ambiguity Detection in Cluster Analysis Using Simulated Annealing 389
E. Mohebi and M.N.M. Sap

Interactive Quality Analysis in the Automotive Industry: Concept and Design of an Interactive, Web-Based Data Mining Application 402
Steffen Fritzsche, Markus Mueller, and Carsten Lanquillon

NARFO Algorithm: Mining Non-redundant and Generalized Association Rules Based on Fuzzy Ontologies 415
Rafael Garcia Miani, Cristiane A. Yaguinuma, Marilde T.P. Santos, and Mauro Biajiz

Automated Construction of Process Goal Trees from EPC-Models to Facilitate Extraction of Process Patterns 427
Andreas Bögl, Michael Schrefl, Gustav Pomberger, and Norbert Weber

Part III: Information Systems Analysis and Specification

A Service Integration Platform for the Labor Market 445
Mariagrazia Fugini

Developing Business Process Monitoring Probes to Enhance Organization Control 456
Fabio Mulazzani, Barbara Russo, and Giancarlo Succi

Text Generation for Requirements Validation 467
Petr Kroha and Manuela Rink

Automatic Compositional Verification of Business Processes 479
Luis E. Mendoza and Manuel I. Capel

Actor Relationship Analysis for the i* Framework 491
Shuichiro Yamamoto, Komon Ibe, June Verner, Karl Cox, and Steven Bleistein

Towards Self-healing Execution of Business Processes Based on Rules ... 501
Mohamed Boukhebouze, Youssef Amghar, Aïcha-Nabila Benharkat, and Zakaria Maamar

Towards Flexible Inter-enterprise Collaboration: A Supply Chain Perspective 513
Boris Shishkov, Marten van Sinderen, and Alexander Verbraeck

A Model-Based Tool for Conceptual Modeling and Domain Ontology Engineering in OntoUML 528
Alessander Botti Benevides and Giancarlo Guizzardi

Concepts-Based Traceability: Using Experiments to Evaluate Traceability Techniques 539
Rodrigo Perozzo Noll and Marcelo Blois Ribeiro

A Service-Oriented Framework for Component-Based Software Development: An i* Driven Approach 551
Yves Wautelet, Youssef Achbany, Sodany Kiv, and Manuel Kolp

A Process for Developing Adaptable and Open Service Systems: Application in Supply Chain Management 564
Yves Wautelet, Youssef Achbany, Jean-Charles Lange, and Manuel Kolp

Business Process-Awareness in the Maintenance Activities 577
Lerina Aversano and Maria Tortorella

BORM-points: Introduction and Results of Practical Testing 590
Zdenek Struska and Robert Pergl

A Technology Classification Model for Mobile Content and Service Delivery Platforms 600
Antonio Ghezzi, Filippo Renga, and Raffaello Balocco

Patterns for Modeling and Composing Workflows from Grid Services . . . 615
Yousra Bendaly Hlaoui and Leila Jemni Ben Ayed

A Case Study of Knowledge Management Usage in Agile Software Projects 627
Anderson Yanzer Cabral, Marcelo Blois Ribeiro, Ana Paula Lemke, Marcos Tadeu Silva, Mauricio Cristal, and Cristiano Franco

A Hierarchical Product-Property Model to Support Product Classification and Manage Structural and Planning Data 639
Diego M. Giménez, Gabriela P. Henning, and Horacio P. Leone

Collaborative, Participative and Interactive Enterprise Modeling 651
Joseph Barjis

Part IV: Software Agents and Internet Computing

e-Learning in Logistics Cost Accounting Automatic Generation and Marking of Exercises 665
Markus Siepermann and Christoph Siepermann

Towards Successful Virtual Communities 677
Julien Subercaze, Christo El Morr, Pierre Maret, Adrien Joly, Matti Koivisto, Panayotis Antoniadis, and Masayuki Ihara

A Multiagent-System for Automated Resource Allocation in the IT Infrastructure of a Medium-Sized Internet Service Provider 689
Michael Schwind and Marc Goederich

AgEx: A Financial Market Simulation Tool for Software Agents 704
Paulo André L. De Castro and Jaime S. Sichman

A Domain Analysis Approach for Multi-agent Systems Product Lines . . . 716
Ingrid Nunes, Uirá Kulesza, Camila Nunes, Carlos J.P. de Lucena, and Elder Cirilo

A Reputation-Based Game for Tasks Allocation 728
Hamdi Yahyaoui

Remote Controlling and Monitoring of Safety Devices Using Web-Interface Embedded Systems 737
A. Carrasco, M.D. Hernández, M.C. Romero, F. Sivianes, and J.I. Escudero

Recognizing Customers' Mood in 3D Shopping Malls Based on the Trajectories of Their Avatars 745
Anton Bogdanovych, Mathias Bauer, and Simeon Simoff

Assembling and Managing Virtual Organizations out of Multi-party Contracts 758
Evandro Bacarin, Edmundo R.M. Madeira, and Claudia Medeiros

A Video-Based Biometric Authentication for e-Learning Web Applications 770
Bruno Elias Penteado and Aparecido Nilceu Marana

Modeling JADE Agents from GAIA Methodology under the Perspective of Semantic Web 780
Ig Ibert Bittencourt, Pedro Bispo, Evandro Costa, João Pedro, Douglas Véras, Diego Dermeval, and Henrique Pacca

A Business Service Selection Model for Automated Web Service Discovery Requirements 790
Tosca Lahiri and Mark Woodman

Part V: Human–Computer Interaction

An Agile Process Model for Inclusive Software Development 807
Rodrigo Bonacin, Maria Cecília Calani Baranauskas, and Marcos Antônio Rodrigues

Creation and Maintenance of Query Expansion Rules 819
Stefania Castellani, Aaron Kaplan, Frédéric Roulland, Jutta Willamowski, and Antonietta Grasso

Stories and Scenarios Working with Culture-Art and Design in a Cross-Cultural Context 831
Elizabeth Furtado, Albert Schilling, and Liadina Camargo

End-User Development for Individualized Information Management: Analysis of Problem Domains and Solution Approaches 843
Michael Spahn and Volker Wulf

Evaluating the Accessibility of Websites to Define Indicators in Service Level Agreements 858
Sinésio Teles de Lima, Fernanda Lima, and Káthia Marçal de Oliveira

Promoting Collaboration through a Culturally Contextualized Narrative Game 870
Marcos Alexandre Rose Silva and Junia Coutinho Anacleto

Applying the Discourse Theory to the Moderator's Interferences in Web Debates 882
Cristiano Maciel, Vinícius Carvalho Pereira, Licinio Roque, and Ana Cristina Bicharra Garcia

ExpertKanseiWeb: A Tool to Design Kansei Website 894
Anitawati Mohd Lokman, Nor Laila Md. Noor, and Mitsuo Nagamachi

Evaluation of Information Systems Supporting Asset Lifecycle Management 906
Abrar Haider

Fast Unsupervised Classification for Handwritten Stroke Analysis 918
Won-Du Chang and Jungpil Shin

Interfaces for All: A Tailoring-Based Approach 928
Vânia Paula de Almeida Neris and Maria Cecília Calani Baranauskas

Integrating Google Earth within OLAP Tools for Multidimensional Exploration and Analysis of Spatial Data 940
Sergio Di Martino, Sandro Bimonte, Michela Bertolotto, and Filomena Ferrucci

An Automated Meeting Assistant: A Tangible Mixed Reality Interface for the AMIDA Automatic Content Linking Device 952
Jochen Ehnes

Investigation of Error in 2D Vibrotactile Position Cues with Respect to Visual and Haptic Display Properties: A Radial Expansion Model for Improved Cuing 963
Nicholas G. Lipari, Christoph W. Borst, and Vijay B. Baiyya

Developing a Model to Measure User Satisfaction and Success of Virtual Meeting Tools in an Organization 975
A.K.M. Najmul Islam

Author Index 989

Part I

Databases and Information Systems Integration

MIDAS: A Middleware for Information Systems with QoS Concerns*

Luís Fernando Orleans and Geraldo Zimbrão

COPPE/UFRJ - Computer Science Department - Graduate School and Research in Engineering – Federal University of Rio de Janeiro
{lforleans,zimbrao}@cos.ufrj.br

Abstract. One of the most difficult tasks in the design of information systems is how to control the behaviour of the back-end storage engine, usually a relational database. As the load on the database increases, the longer issued transactions will take to execute, mainly because the presence of a high number of locks required to provide isolation and concurrency. In this paper we present MIDAS, a middleware designed to manage the behaviour of database servers, focusing primarily on guaranteeing transaction execution within an specified amount of time (deadline). MIDAS was developed for Java applications that connects to storage engines through JDBC. It provides a transparent QoS layer and can be adopted with very few code modifications. All transactions issued by the application are captured, forcing them to pass through an Admission Control (AC) mechanism. To accomplish such QoS constraints, we propose a novel AC strategy, called 2-Phase Admission Control (2PAC), that minimizes the amount of transactions that exceed the established maximum time by accepting only those transactions that are not expected to miss their deadlines. We also implemented an enhancement over 2PAC, called *diffserv* – which gives priority to small transactions and can adopted when their occurrences are not often.

Keywords: Database Performance, QoS for Databases, Transactions with deadlines, Midas.

1 Introduction

Information systems are usually designed with a multi-tier architecture, each tier being responsible for a specific function. Commonly, there are at least 3 tiers, comprising presentation, application and persistence logics (see figure 1). In a *web* information system, for instance, the first tier contains web pages (static or dynamic) that are displayed to clients. The second comprises business rules and constraints, validating and/or processing users data input. Finally, the database is responsible for storage and retrieval of such data.

Although all tiers present potential performance bottlenecks, the database layer is commonly the most problematic, being responsible for performance degradation in peak situations. Intuitively, this can be explained by the high number of disk accesses

* This work was partially financed by CNPq – Brazil.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 3–13, 2009.
© Springer-Verlag Berlin Heidelberg 2009

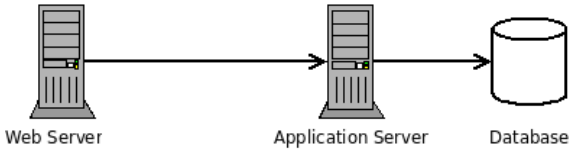


Fig. 1. A typical 3-tier architecture used in information systems

necessary for both read and write operations performed by the database. Furthermore, in databases, all data modifications occur within *transactions*, that must accomplish the well known ACID properties. The isolation (letter "I" of ACID) constraint avoids interleaved executions of transactions (partially or fully) by controlling concurrency. As part of such a control, all transactions must acquire a *lock* to the data prior update operations. If it is not possible to acquire a lock (i.e., data are being used by another transaction), the transaction must wait until its release. The concurrency mechanism is also responsible for performance degradations of the database tier and, consequently, the whole system.

This paper presents a middleware named MIDAS that was designed to control the behaviour of database servers. By using admission control, dynamic transaction classification and other features, it keeps the average response time always below an specified amount of time, guaranteeing QoS constraints that could be established through Service Level Agreement (SLA). MIDAS was developed for Java applications that use JDBC to connect to databases.

Basically, the middleware works as follows: every issued transaction (tx) is intercepted by MIDAS. Then, MIDAS checks if tx can be immediately executed by consulting if the number of tasks being executed is lower than an specified threshold (denoted here as multiprogramming level or MPL, in short). If the maximum MPL had not been reached, the transaction is executed. Otherwise, its estimated execution time is computed (different policies can be used for handling transactions with different estimated durations but, for now, we should assume that they all are handled the same way) and an *overflow strategy* is applied, e.g. put the transaction in a waiting queue.

Such strategy is crucial for controlling the behaviour of DBMS servers, as well for guaranteeing time-constraints. Our experimental results show that it is possible to achieve both goals with MIDAS.

1.1 Contributions

The main contributions of this paper are:

1. A middleware that concerns about keeping the load on database servers under control;
2. Also, a middleware that is both easy to adopt and to extend;
3. A basic strategy to classify transactions according to their durations;
4. A novel admission control policy, called 2PAC. We also present an enhancement over this policy, through the concept of *differentiation of services* (diffserv);

The remaining of this paper is structured as follows: sections 2 and 3 explain the middleware architecture, basic services provided and how the classification mechanism works. Sections 4 and 5 present experiments we have made and a discussion about the obtained results, while section 6 gives the related work and background in which this paper is based on Finally, section 7 lists the conclusions, pointing future directions for this work.

2 MIDAS Architecture

MIDAS is a middleware designed for Java applications that accesses databases through the Java Database Connectivity (JDBC) API. In fact, it does not require a *relational database*, but any persistence mechanism that can be accessed through a JDBC driver. Figure 2 presents a simplified class diagram of MIDAS architecture. As can be noticed, MIDAS makes use of the *Proxy* Design Pattern [10] as the mechanism to intercept the requests sent by users to the database.

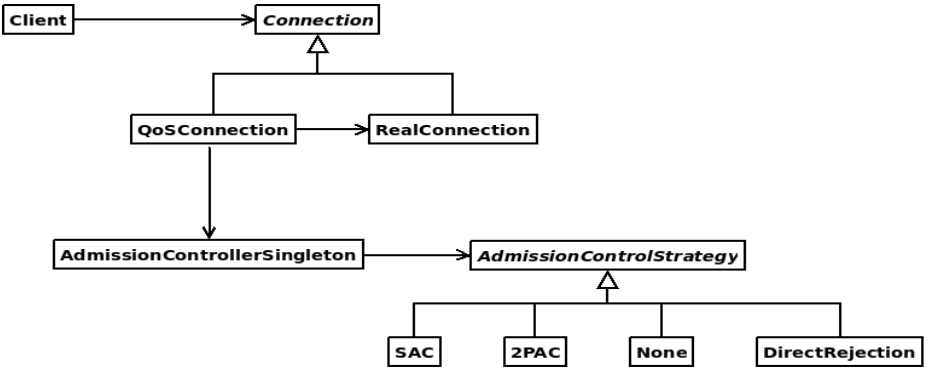


Fig. 2. Simplified class-diagram of MIDAS

At the beginning of any transaction, the *AdmissionControlSingleton* (ACS) comes into play. Firstly, the ACS checks if the maximum MPL had been reached and, if not, the transaction goes directly to execution. Otherwise, the ACS estimates the transaction duration (see section 3 for more details on how MIDAS does such an estimation) then, it passes both transaction and its expected estimation to a previously defined *Admission Control Policy* (ACP). There are four ACPs already defined for MIDAS:

- *None*: in practice, this is a non-admission control policy, also known as *best-effort.* All issued transactions are executed independently of maximum MPL had been reached or not;
- *Direct Rejection*: there is no FCFS queue. All transactions that exceed the MPL are immediately rejected.
- *Simple Admission:* where all transactions that exceeds the MPL are forwarded to an FCFS queue. It has the inconvenient that for high arrival rates transactions

might have to wait a long period (perhaps even longer than the deadline) before execution – because the queue might be extremely long;

- *2 Phase Admission Control (2PAC)*: this approach is quite similar to SAC. The main difference relies on the use of the transactions' estimated execution times to manage the queue size. In short, 2PAC uses the sum of all estimated durations on the queue as the *minimum time* for the queue be fully consumed. If this minimum time is greater than a threshold, then every new transaction is rejected by the system;

Figure 3 shows how SAC and 2PAC behave.

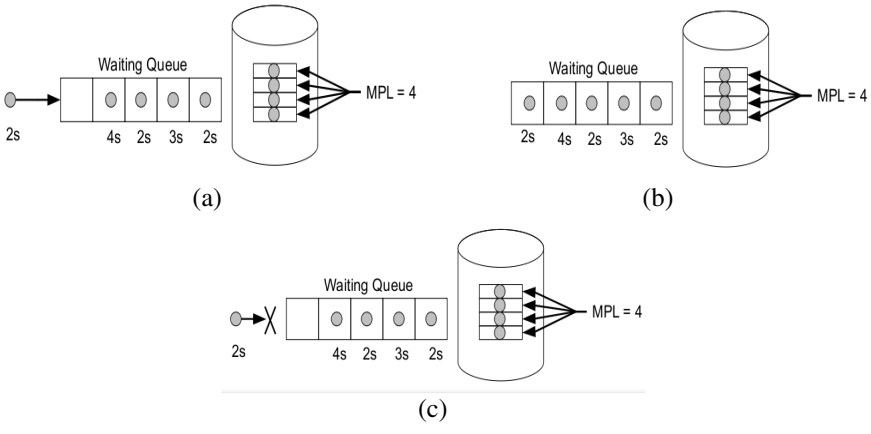


Fig. 3. Behaviours of Admission Control policies: a transaction arrives (a) with a 11-seconds deadline. In SAC, it is accepted (b). In 2PAC, transaction is rejected (c).

The definition about which ACP MIDAS should use is up to the user and can be set in its *configuration file.*

We have implemented another enhancement over the 2PAC algorithm: diffserv, which is an acronym for differentiation of services. It is a fundamental building block for QoS networks in the sense that it gives some kind of priority to more critical packets, e.g., video or audio streaming packets. This concept can be applied also in information systems, where some transactions may be prioritized. According to the related work [14], in a distributed or parallel environment, it is feasible to give priority to short transactions without overwhelming the throughput of the big transactions.

In this work, the short transactions can pass through the admission control mechanisms if the diffserv flag is properly set. Intuitively, this can be a reasonable choice in a 1-server configuration only when the number of big transactions is much higher than the number of small ones – otherwise it can degrade the performance by letting lots of small tasks execute at the same time.

2.1 Code Modifications

As MIDAS acts like a Proxy, it does not require any deep source code modifications prior its adoption. Both `QoSConnection` and `RealConnection` (see figure 2) implement the `java.sql.Connection` interface. Thus, the most significant modifications are:

1. How the connection object is created. A `QoSConnection` object should be instantiated using the `new` operator;
2. Before each transaction initiation, a *transaction name* must be informed to MIDAS. This serves to build a map with the estimated execution times.

So, the following code snippet

```
Class.forName("driverClassName");
Connection c = DriverManager.getConnection(dbProps);
Statement stmt = c.createStatement();
stmt.execute(sql);
```

should be changed to:

```
Connection c = new QoSConnection();
Statement stmt = c.createStatement();
((QoSStatement)stmt).setName(name);
stmt.execute(sql);
```

Because MIDAS will create the real connection to the database, all parameters needed to perform this task (JDBC driver class name, connection URL, user name and user password) should be properly described in the configuration file.

3 Tx Estimation

The most important characteristic of MIDAS is its ability to estimate execution times of issued transactions. Based on this, it is possible to manage queue size and minimize the number of transactions that will take longer than an specified amount of time. This section explains how MIDAS estimates execution times.

As mentioned in the previous section, before issuing a transaction, users must inform *transaction's name* to the middleware. This serves to build a map as follows:

{key value = TxName; value = pounded mean time}

The map keeps a pounded mean comprising the past response times of the corresponding transaction. It is pounded because it is necessary to give higher weights to recent executions – as we want to reflect the *recent behaviour* of the database server. So, the basic formula used to calculate the pounded mean was:

$$E(N)=0.4*Tx(N-1)+... +0.1*Tx(N-4) \quad \text{(i)}$$

Where *Tx(N-1)* represents the response time of the $N\text{-}1^{th}$ executed transaction.

Although simple, this estimation perfectly addressed our needs to compute queue durations, as stated in the experiments results (section 6).

4 Experiments

Through the experiments we could evaluate two aspects of MIDAS:

1. How easy it is its adoption; and
2. Compare the performances of two Admission Control Policies present in MIDAS (SAC and the novel algorithm 2PAC), indicating their strengths and typical scenarios where they should be successfully used. We also compared the performances of 2PAC with and without the diffserv enhancement.

As target system (the one in which we want to adopt MIDAS) we used the jTPCC system (http://jtpcc.sourceforge.net/), which is an open-source implementation of the TPC-C benchmark (http://www.tpc.org/tpcc/).

All experiments were executed using a Pentium 4 3.2GHz, with 2GB RAM DDR2 and a 200 GB SATA HD which was responsible for creating the threads that simulate the clients. The server was a Pentium II MMX 350MHz, with 256MB RAM and a 60GB IDE HD. Both computers were running a Debian Linux, with Kernel version 2.6 and were connected by a full-duplex 100Mbps Ethernet link. A PostgreSQL 8.1 database server was running on the server machine and the database size was 1.11GB. The client machine used a Sun Microsystems' Java Virtual Machine, version 1.5. The database was created with 10 warehouses, which allows a maximum of 100 terminals (emulated clients) by the TPC-C specification.

The reason for using the slower computer as the database server relies on our need to stress the system. Our intention was not to maximize the throughput within deadline (TWD), but to compute the difference between TWD and throughput for each MPL value.

4.1 Workload Composition

In order to effectively test the performance of MIDAS, we used 2 transaction mixes and 2 load scenarios, leading up to 4 different workload compositions. The details of each are given below. The default transaction mix is the TPC-C default mix, while the heavy-tailed alternative depicts a *typical scenario* where short transactions represent 95% of the system load, similar to the observations contained in [11].

5 Results

As described in section 2.1 the code modifications necessary prior utilization of MIDAS were quite simple. As the name of transactions used in TPC-C are already defined in its specification (*new order, payment, delivery, stock level and order status*), we used them to build the estimated durations map. As expected, no further modifications were required. In fact, in a huge enterprise system, with thousands of lines of code this modification can be more troublesome – although such effort can be significantly reduced if the connection to the database is established through *Factory*

Table 1. Transaction mixes

Transaction Mix	Transactions Occurrences
Heavy-Tailed (used for comparison purposes)	●Delivery: 5% ●Stock-Level: 95% ●Other transactions: 0%
Default	●New Order: 45% ●Payment: 43 % ●Other transactions: 4%

Table 2. Think times

Load Type	Think Time
Medium-Load	Exponential distribution, with mean 8 seconds and a maximum of 80 seconds
High-Load	Exponential distribution, with mean 4 seconds and a maximum of 40 seconds.

Design Pattern, which would encapsulate all the logic. Hence, the change would be altering a single method of one class.

The performance results show how beneficial can be the adoption of an engine like MIDAS. We utilized two metrics for compare the gains: simple throughput and throughput within deadline, which represents the rate of transactions that ended within the deadline.

For the first round of experiences, a heavy-tailed scenario was used. The results are displayed in figures 4a and 4b. Since the number of short transactions is much greater than the number of long transactions, no diffserv was necessary. The first thing to be noticed from the graphics is that the throughput within deadline (TWD – number of transactions that ended within the deadline per minute) for the SAC case is much smaller than the total throughput (TT – total number of transactions that ended per minute) for the medium load case. For the high load scenario, the TWD is even worse: no transaction ended within the deadline at all! So it turns out the necessity of a new approach, one which tries to avoid deadline misses. The graphic in figure 4a shows the effectiveness of the 2PAC approach, since the TWD is always close to TT.

In figure 4b, we can see that the maximum throughput is reached with only 2 MPL by the system with 2PAC. This occurs because the workload is comprised mostly by short transactions, which execute very fast. As the MPL increases, the TT keeps almost unaltered, while TWD decreases. Two conclusions can be taken from the last statement: 1) 2PAC is more robust than SAC, since SAC has its performance deeply degraded by higher MPL values; and 2) the workload variability is responsible for the degradation of TWD of 2PAC in figure 4b for higher MPL values, since more long transactions get to execute.

In the second round of experiments, we used the default transaction mix, established by the TPC-C specification. Figures 5a and 5b show the graphics for medium load and high load, respectively. Again, it turns out that 2PAC mechanism is

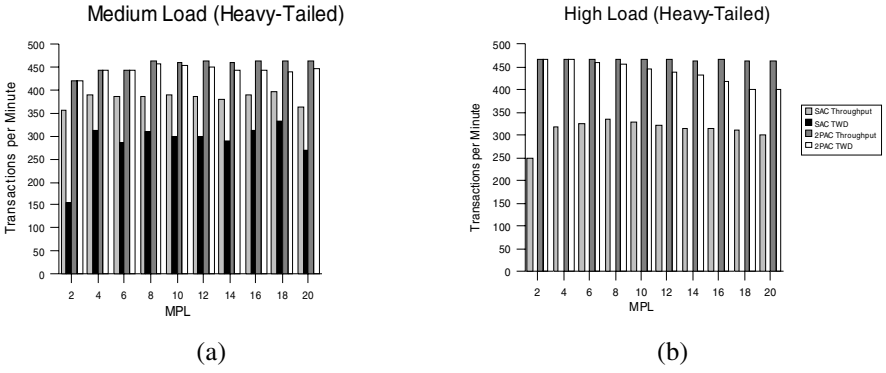


Fig. 4. Performance comparison for heavy-tailed workloads with medium (a) and high-loads (b)

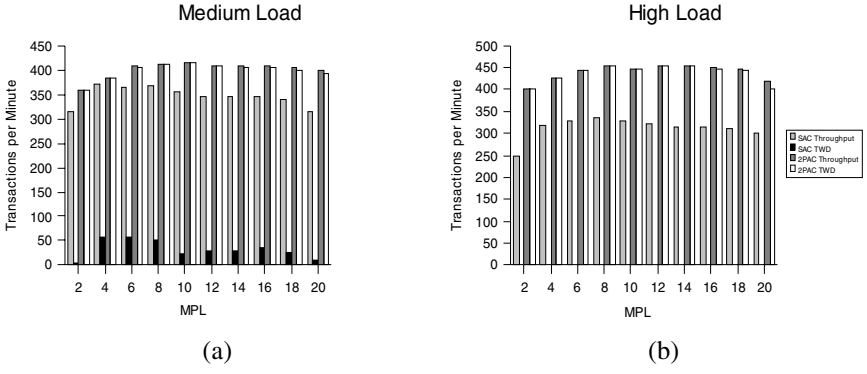
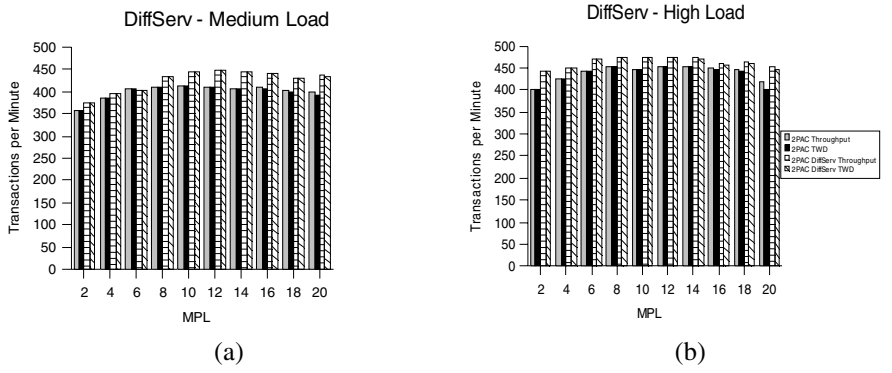


Fig. 5. Performance comparison for TPC-C's default workload with medium (a) and high-loads (b)

much more robust than SAC, which can be attested by the comparison of TWD of both methods: 2PAC's TWD is almost the same as TT for all MPL values, in SAC these values of TWD are very low (0 for the high load scenario). In this transaction mix, the number of short transactions is much smaller than the number of big transactions, so we were able to use the diffserv engine as another enhancement to the system The use of diffserv also increased the performance, but its contribution is smaller than the isolated use of 2PAC.

Figure 5a shows that SAC has a very small TWD, due to the massive presence of long transactions, achieving its highest value (58 transactions per minute) with MPL 6. On the other hand, the 2PAC approach has as maximum TWD 414 transactions per minute, more than 7 times higher than SAC! When the diffserv flag is set, then the TWD goes to 447 transactions per minute (figure 6a).

The graphic 5b shows that no matter how the load on the system increases, the TT of all techniques remains almost unaltered. But, in SAC case, the queue grows uncontrolled and the time a transaction spends waiting for execution makes it miss the deadline. In fact, TWD is zero for all MPL values in SAC. Again, the 2PAC has a better performance, keeping TWD close to TT, reaching its maximum value at 14



(a) (b)

Fig. 6. Performance comparison between 2PAC and 2PAC + DiffServ enhancement with medium (a) and high-loads (b)

MPL with 453 transactions per minute. With diffserv enhancement, the maximum TWD is reached with MPL values between 8 and 12, with 474 transactions per minute, in figure 6b.

6 Related Work

In [6] the authors propose session-based Admission Control (SBAC), noting that longer sessions may result in purchases and therefore should not be discriminated in overloaded conditions. They propose self-tunable admission control based on hybrid or predictive strategies. Reference [5] uses a rather complex analytical model to perform admission control. There are also approaches proposing some kind of service differentiation: [3] proposes architecture for Web servers with differentiated services. Reference [15] proposes an approach for Web Servers to adapt automatically to changing workload characteristics and [9] proposes a strategy that improves the service to requests using statistical characterization of those requests and services.

In [14], it is proposed a dynamic load-balancing algorithm, called ORBITA, that tries to guarantee deadlines by applying some kind of DiffServ, where small tasks have priorities and execute on a dedicated server. The big tasks have to pass through the admission control mechanism and can be rejected, if the maximum MPL (calculated in runtime) had been reached.

Reference [16] studies how to obtain a maximum throughput with the lowest MPL value using a simple admission control approach, while [9] and [1] proposes the C-JDBC middleware, which offers high-availability and scalability issues through a Proxy on the Java connection. However, it is focused on *distributed* or *parallel databases*, whereas the focus of MIDAS is guarantee deadlines for transactions on *centralized databases.*

Comparing to our own work, none of the previously mentioned intended to study how to manage the growth of the waiting queue, which has its size dynamically computed according to the workload characteristics, in order to accept only the tasks that will be able to execute within the deadline. Neither did the related work propose a usable middleware for centralized databases, offering QoS concerns.

7 Conclusions and Future Works

Stressed database servers may use an admission control mechanism to achieve better throughput. This becomes a problem when transactions have deadlines to meet as traditional admission control (SAC) models (with a FCFS waiting queue) may not be applied, since the queue time is a potential point for QoS failures. This paper presented a middleware named MIDAS that was designed to keep the behaviour of database servers accessed through JDBC under control. The main idea of MIDAS relies on the use of the Proxy Design Pattern to offer information systems QoS capabilities – without requiring deep source code modifications. Our work also presented the 2-Phase Admission Control (2PAC) algorithm, which estimates the execution time of a transaction according to a mathematical formula that takes into account the last 4 execution times of the same transaction. Once the execution time is estimated, it is possible to calculate how long the transaction will spend on the waiting queue and, furthermore, if the transaction will be able to be completed before the deadline. If the middleware calculates that the transaction would miss the deadline, it is rejected by the system.

Then, we altered an existing information system (an open-source implementation of the TPC-C benchmark) and we could attest that very few code modifications were necessary to make use of MIDAS. We ran the benchmark with 4 different workload compositions using 2 admission control strategies offered by MIDAS: SAC and 2PAC. A last enhancement, diffserv, was included in the experiments for the default workload. Diffserv gives priority for short transactions, letting them pass through the admission control mechanism.

The results showed that, in order to reach a good rate of transactions ended within deadline, it is necessary to limit the number of transactions on the waiting queue. This way, all transactions that are supposed to miss their deadlines are rejected. Despite the high number of transactions being rejected, the improvement on system's performance is almost 8 times higher when both 2PAC and DiffServ enhancements are used.

As future works, we intend to investigate how a multi-server environment is affected by admission control policies and try to establish a connection between them, leading to a complete highly scalable, distributed database system. Also, we intend to investigate how to effectively identify and estimate the duration of transactions. The solution adopted in this paper (using a map with pounded mean response times) was sufficient for what this work was intended, but we are concerned on how to generalize such a concept for a system with ad-hoc transactions. We also intend to work on database internals level and study the viability of adding time-constraints mechanisms to queries and/or transactions.

References

1. Amza, C., Cox, A.L., Zwaenepoel, W.: A Comparative Evalution of Transparent Scaling Techniques for Dynamic Content Servers. In: ICDE 2005 International Conference On Data Engineering (2005)
2. Barker, K., Chernikov, A., Chrisochoides, N., Pingali, K.: A Load Balancing Framework for Adaptive and Asynchronous Applications. IEEE Transactions on Parallel and Distributed Systems 15(2) (2004)

3. Bhatti, N., Friedrich, R.: Web server support for tiered services. IEEE Network 13(5), 64–71 (1999)
4. Cardellini, V., Casalicchio, C.M., Yu, P.S.: The State of the Art in Locally Distributed Web-Server Systems. ACM Computing Surveys 34, 263–311 (2002)
5. Chen, X., Mohapatra, P., Chen, H.: An admission control scheme for predictable server response time for Web accesses. In: WWW 2002, World Wide Web Conference, Hong Kong (2002)
6. Cherkasova, Phaal: Session-based admission control: A mechanism for peak load management of commercial Web sites. IEEE Req. on Computers 51(6) (2002)
7. Crovella, M., Bestavros, A.: Self-similarity in World Wide Web traffic: Evidence and possible causes. IEEE/ACM Transactions on Networking, 835–836 (1999)
8. Dyachuk, D., Deters, R.: Optimizing Performance of Web Service Providers. In: International Conference on Advanced Information Networking and Applications, Niagara Falls, Ontario, Canada, pp. 46–53 (2007)
9. Elnikety, S., Nahum, E., Tracey, J., Zwaenepoel, W.: A Method for Transparent Admission Control and Request Scheduling in E-Commerce Web Sites. In: World Wide Web Conference, New York City, NY, USA (2004)
10. Gamma, E., et al.: Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley, Reading (1994)
11. Harchol-Balter, M., Downey, A.: Exploiting process lifetime distributions for dynamic load-balancing. ACM Transactions on Computer Systems (1997)
12. jTPCC, Open-source Java implementation of TPC-C benchmark, `http://jtpcc.sourceforge.net/`
13. Knightly, E., Shroff, N.: Admission Control for Statistical QoS: Theory and Practice. IEEE Network 13(2), 20–29 (1999)
14. Orleans, L.F., Furtado, P.N.: Fair load-balance on parallel systems for QoS. In: International Conference on Parallel Programming, Xi-An, China (2007)
15. Pradhan, P., Tewari, R., Sahu, S., Chandra, A., Shenoy, P.: An observation-based approach towards self managing Web servers. In: International Workshop on Quality of Service, Miami Beach, FL (2002)
16. Schroeder, B., Harchol-Balter, M.: Achieving class-based QoS for transactional workloads. In: International Conference on Data Engineering, p. 153 (2006)
17. Serra, A., Gaïti, D., Barroso, G., Boudy, J.: Assuring QoS Differentiation and Load-Balancing on Web Servers Clusters. In: IEEE Conference on Control Applications, vol. 8, pp. 85–890 (2005)
18. TPC-C Benchmark Homepage, `http://www.tpc.org/tpcc/`

Instance-Based OWL Schema Matching

Luiz André P. Paes Leme, Marco A. Casanova,
Karin K. Breitman, and Antonio L. Furtado

Department of Informatics – Pontifical Catholic University of Rio de Janeiro
Rua Marquês de S. Vicente, 225 – Rio de Janeiro, RJ – Brazil CEP 22451-900
{lleme,casanova,karin,furtado}@inf.puc-rio.br

Abstract. Schema matching is a fundamental issue in many database applications, such as query mediation and data warehousing. It becomes a challenge when different vocabularies are used to refer to the same real-world concepts. In this context, a convenient approach, sometimes called extensional, instance-based or semantic, is to detect how the same real world objects are represented in different databases and to use the information thus obtained to match the schemas. This paper describes an instance-based schema matching technique for an OWL dialect. The technique is based on similarity functions and is backed up by experimental results with real data downloaded from data sources found on the Web.

Keywords: Schema matching, OWL, Similarity functions.

1 Introduction

A *database conceptual schema*, or simply a *schema*, is a high level description of how database concepts are organized. A *schema matching* from a source schema *S* into a target schema *T* defines concepts in *T* in terms of the concepts in *S*.

The problem of finding a schema matching becomes a challenge when different vocabularies are used to refer to the same real-world concepts [6]. In this case, a convenient approach, sometimes called extensional, instance-based or semantic, is to detect how the same real-world objects are represented in different databases and to use the information thus obtained to match the schemas. This approach is grounded on the interpretation, traditionally accepted, that "terms have the same extension when true of the same things" [14].

We address in this paper the problem of matching two schemas that belong to an expressive OWL dialect. We adopt an instance-based approach and, therefore, assume that a set of instances from each schema is available.

The major contributions of this paper are three-fold. First, we decompose the problem of OWL schema matching into the problem of vocabulary matching and the problem of concept mapping. We also introduce sufficient conditions guaranteeing that a vocabulary matching induces a correct concept mapping. Second, we describe an OWL schema matching technique based on the notion of similarity. Third, we evaluate the precision of the technique using data available on the Web.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 14–26, 2009.
© Springer-Verlag Berlin Heidelberg 2009

Rahm and Bernstein [15] is an early survey of schema matching techniques. Euzenat and Shvaiko [9] survey ontology matching techniques. Castano et al. [7] describe the H-Match algorithm to dynamically match ontologies.

Bilke and Naumann [1] describe an instance-based technique that explores similarity algorithms. Brauner et al. [2] adopt the same idea to match two thesauri. Wang et al. [16] describe a technique to match Web databases, which uses a set of typical instances. Brauner et al. [4] apply this idea to match geographical database. Brauner et al. [3] describe a matching algorithm based on measuring the similarity between attribute domains.

Unlike any of the above instance-based techniques, the matching process we describe uses similarity functions to induce vocabulary matchings in a non-trivial way, coping with an expressive OWL dialect. We also illustrate, through a set of examples, that the structure of OWL schemas may lead to incorrect concept mappings and indicate how to avoid such pitfalls.

This paper is organized as follows. Section 2 introduces the OWL dialect adopted and the notions of vocabulary matching and concept mapping. Section 3 describes our technique to obtain OWL schema matchings. Section 4 contains experimental results. Finally, Section 5 lists the conclusions and directions for future work.

2 OWL Schema Matching

2.1 OWL Extralite

We assume that the reader is familiar with basic XML concepts. In particular, recall that a *resource* is anything identified by an URIref and that an XML *namespace* or a *vocabulary* is a set of URIrefs. A *literal* is a character string that represents an XML Schema datatype value. We refer the reader to [5] for the details.

An *RDF statement* (or simply a *statement*) is a triple *(s,p,o)*, where *s* is a URIref, called the *subject* of the statement, *p* is a URIref, called the *property* of the statement, and *o* is either a URIref or a literal, called the *object* of the statement; if *o* is a literal, then *o* is also called the *value* of property *p*.

The *Web Ontology Language* (OWL) describes classes and properties in a way that facilitates machine interpretation of Web content. The description of OWL is organized as three dialects: OWL Lite, OWL DL and OWL Full.

We will work with an OWL dialect, that we call *OWL Extralite*. It supports named classes, datatype and object properties, subclasses, and individuals. The *domain* of a datatype or object property is a class, the *range* of a datatype property is an XML schema type, whereas the *range* of an object property is a class. As *property restrictions*, the dialect admits minCardinality and maxCardinality, with the usual meaning. As *property characteristic*, it allows just the InverseFunctionalProperty, which captures simple keys. We note that only OWL Full supports the InverseFunctionalProperty for datatype properties.

An *OWL schema* (more often called an *OWL ontology*) is a collection of RDF triples that use the OWL vocabulary. A *concept* of an OWL schema is a class, datatype property or object property defined in the schema. The *vocabulary* of the schema is the set of concepts defined in the schema (a set of URIrefs). The scope of a property name is global to the OWL schema, and not local to the class indicated as its domain.

A triple of the form *(s,*`rdf:type`*,c)* indicates that *s* is an instance of a class *c*; a triple of the form *(s,p,v)* indicates that *s* has a datatype property *p* with value *v*; and a triple of the form *(s,p,o)* indicates that *s* and *o* are related by an object property *p*.

In the rest of the paper, we refer an to OWL Extralite schema simply as a *schema*.

Figures 1 and 2 show schemas for fragments of the Amazon and the eBay databases, using a simplified notation to save space and improve readability. Consistently with XML usage, from this point on, we will use the namespace prefixes `am:` and `eb:` to refer to the vocabularies of the Amazon and the eBay schemas, and qualified names of the form `V:T` to indicate that `T` is a term of the vocabulary `V`.

In Figure 1, for example, `am:title` is defined as a datatype property with domain `am:Product` and range `string` (an XML Schema data type), `am:Book` is declared as a subclass of `am:Product`, and `am:publisher` is defined as an object property with domain `am:Book` and range `am:Publ`. Note that the scope of `am:title` and `am:publisher` is the schema, and not the classes defined as their domains.

Furthermore, although not indicated in Figure 1, we assume that all properties, except `am:author`, have maxCardinality equal to 1, and that `am:isbn` is inverse functional. This means that all properties are single-valued, except `am:author`, which is multi-valued, and that `am:isbn` is a key of `am:Book`. Likewise, although not shown in Figure 2, all properties, except `eb:author`, have maxCardinality equal to 1, and `eb:isbn-10` and `eb:isbn-13` are inverse functional.

Finally, to express concept mappings, we adopt the Semantic Web Rule Language (SWRL) [10], but use a datalog-like syntax to improve readability and save space. An example of an SWRL rule in our simplified syntax would be:

```
Product
 title range string
 listPrice range decimal
 currency range string
Book is-a Product
 author range string
 edition range integer
 isbn range string
 ean range string
 detailPageURL range anyURI
 publisher range Publ
Publ
 name range string
 address range string
Music is-a Product
Video is-a Product
PCHardware is-a Product
```

Fig. 1. An OWL schema for a fragment of the Amazon Database

```
Seller
 name           range string
 redistrationDate range dateTime
 offers range Offer
Offer
 quantity range integer
 startPrice range double
 currency range string
 seller range Seller
 product range Product
Product
 title range string
 condition range string
 returnPolicyDetails range string
 offers range Offer
Book is-a Product
 author range string
 edition range integer
 publicationYear range integer
 isbn-10 range integer
 isbn-13 range integer
 publisher range string
 binding range string
 condition range string
Music is-a Product
DVDMovies is-a Product
   ComputerNetworking is-a Product
```

Fig. 2. An OWL schema for a fragment of the eBay Database

```
eb:publisher(b,n) ← am:publisher(b,p), am:name(p,n)        (1)
```

which says that, if `b` and `p` are related by `am:publisher`, and `p` and `n` by `am:name`, then `b` and `n` are related by `eb:publisher`.

2.2 Vocabulary Matchings and Concept Mappings

We decompose the problem of schema matching into the problem of *vocabulary matching* and the problem of *concept mapping*. In this section, we introduce both notions with the help of examples.

In what follows, let S and T be two schemas, and $\boldsymbol{V}_S$ and $\boldsymbol{V}_T$ be their vocabularies, respectively. Let $\boldsymbol{C}_S$ and $\boldsymbol{C}_T$ be the sets of classes and $\boldsymbol{P}_S$ and $\boldsymbol{P}_T$ be the sets of datatype or object properties in $\boldsymbol{V}_S$ and $\boldsymbol{V}_T$, respectively.

A *contextualized vocabulary matching* between S and T is a finite set μ of quadruples (v_1,e_1,v_2,e_2) such that

- if $(v_1,v_2)\in \boldsymbol{C}_S\times\boldsymbol{C}_T$, then e_1 and e_2 are the top class $\top$
- if $(v_1,v_2)\in \boldsymbol{P}_S\times\boldsymbol{P}_T$, then e_1 and e_2 are classes in $\boldsymbol{C}_S$ and $\boldsymbol{C}_T$ that must be subclasses of the domains, or the domains themselves, of properties v_1 and v_2, respectively

If $(v_1,e_1,v_2,e_2)\in\mu$, we say that μ *matches* v_1 *with* v_2 *in the context of* e_1 *and* e_2, that e_i *is the context of* v_i and that (e_i,v_i) is a *contextualized concept*, for $i=1,2$.

Let $\mathcal{Q}$ be an OWL query or rule language that supports the definition of classes and properties. In general, a *concept mapping* from S into T in $\mathcal{Q}$ is a set γ of expressions of $\mathcal{Q}$ that define concepts in T in terms of the concepts of S. Schemas S and T are called the *source* and the *target* of the concept mapping.

To detect when two instances denote the same real-world object, we need a third notion. Let $\boldsymbol{U}_S$ and $\boldsymbol{U}_T$ be sets of triples of S and T, respectively. An *instance matching* from S into T is a set μ_I of quadruples such that, if $(I,C,J,D)\in\mu_I$, then there are triples $(I,\texttt{rdf:type},C)\in\boldsymbol{U}_S$ and $(J,\texttt{rdf:type},D)\in\boldsymbol{U}_T$. We say that an instance I of a class C in $\boldsymbol{U}_S$ *matches* an instance J of a class D in $\boldsymbol{U}_T$ iff $(I,C,J,D)\in\mu_I$.

The following examples use the eBay and the Amazon schemas of Figures 1 and 2.

Example 1. Table 1 shows an example of a matching between the vocabularies of the eBay and the Amazon schemas. For example, line 1 indicates that classes `am:Book` and `eb:Book` match, in the sense that a triple `(I,rdf:type,am:Book)` that defines `I` as an instance of `am:Book` may be reinterpreted as a triple `(I,rdf:type,eb:Book)` that defines `I` as an instance of `eb:Book`.

Line 2 indicates that properties `am:title` and `eb:title` match only when their domains are restricted to `am:Book` and `eb:Book`, respectively. Thus, a triple `(I,am:title,t)` that defines `t` as a value of `am:title` may be reinterpreted as a triple `(I,eb:title,t)` that defines `t` as a value of `eb:title`, provided that `I` is an instance of `am:Book`.

In the next three examples, suppose that one wants to generate a concept mapping from the Amazon schema (the source schema) into the eBay schema (the target schema), using the vocabulary matching of Table 1.

Table 1. Example of a vocabulary matching

Amazon		eBay	
am:Book	⊤	eb:Book	⊤
am:title	am:Book	eb:title	eb:Book
am:author	am:Book	eb:author	eb:Book
am:listPrice	am:Product	eb:startPrice	eb:Offer
am:name	am:Publ	eb:publisher	eb:Book

Example 2. Line 1 of Table 1 indicates that `am:Book` matches `eb:Book`. It induces a mapping from `am:Book` into `eb:Book` expressed by the rule

$$\texttt{eb:Book(n)} \leftarrow \texttt{am:Book(n)} \tag{2}$$

From Figures 1 and 2, we have that `am:Book` is a subclass of `am:Product` and that `eb:Book` is a subclass of `eb:Product`. However, Table 1 does not indicate that `am:Product` matches `eb:Product`. Therefore, we must include an additional rule to guarantee that a consistent mapping is generated (see Section 2.3)

$$\texttt{eb:Product(n)} \leftarrow \texttt{am:Book(n)} \tag{3}$$

The mappings in (2) and (3) should be understood as follows. Let Q be a query over the eBay schema and assume that Q refers to `eb:Book`. Then, Q will be partly translated to the Amazon schema by replacing `eb:Book` by `am:Book`. Likewise, if Q refers to `eb:Product`, then Q will be partly translated by replacing `eb:Product` by `am:Book`. This means that, if Q asks for products, the translated query Q' will return only books.

Example 3. Consider line 2 of Table 1. Since `am:Book` is the context of `am:title`, the following rule expresses a correct mapping from `am:title` into `eb:title`

$$\texttt{eb:title(b,n)} \leftarrow \texttt{am:title(b,n), am:Book(n)} \tag{4}$$

The mapping in (4) should be understood as follows. Let Q be a query over the eBay schema and assume that Q refers to `eb:title`. Then, Q will be partly translated to the Amazon schema by replacing `eb:title` by

$$\texttt{am:title(b,n) and am:Book(n)}$$

This means that, if Q asks for product titles, for example, the translated query Q' will return only book titles from the Amazon schema, since Q' has an extra restriction

$$\texttt{...and am:Book(n)}$$

Note that the right-hand side of the rule in (4) does not contain the context `eb:Book`, which will only be used when creating a concept mapping from the eBay schema into the Amazon schema.

Example 4. From Table 1 and Figures 1 and 2, we have:

`am:name` matches `eb:publisher` (5)

`am:Publ` and `eb:Book` are the domains of `am:name` and `eb:publisher` (6)

`am:Publ` does not match `eb:Book` (7)

`am:Book` matches `eb:Book` (8)

From (6), we cannot directly map `am:name` into `eb:publisher`. Indeed, the rule

```
eb:publisher(b,n) ← am:name(b,n)                    (9)
```

expresses an incorrect mapping since `b` on the right-hand side stands for an instance of `am:Publ` (the domain of `am:name`), whereas `b` on the left-hand side stands for an instance of `eb:Book` (the domain of `eb:publisher`), but Table 1 does not indicate that `am:Publ` matches `eb:Book`. By contrast, the rule

```
eb:publisher(b,n) ← am:publisher(b,p), am:name(p,n)    (10)
```

is a correct mapping. Observing the right-hand side of the rule, we have that `b` stands for an instance of `am:Book`, which the object property `am:publisher` associates with an instance `p` of `am:Publ`, and the datatype property `am:name` in turn associates `p` with a string `n`. Now, observing the left-hand side of the rule, the datatype property `eb:publisher` associates `b`, an instance of `am:Book`, reinterpreted as an instance of `eb:Book`, (the domain of `eb:publisher`) with `n`. This reinterpretation is consistent, since Table 1 also indicates that `am:Book` matches `eb:Book` (see Example 1).

2.3 Consistent OWL Matchings

We briefly discuss in this section the consistency of OWL Extralite vocabulary matchings, referring the reader to [12] for the detailed definitions and proofs.

In what follows, we use the notion of subsumption as in Description Logic. We say that a class c *dominates* a class d iff there is a sequence $(c_1,c_2,...,c_n)$ of classes such that $c=c_1$, $d=c_n$ and, for each $i \in [1,n-2)$, either c_{i+1} is declared as a subclass of c_i or there is an object property whose domain is c_i and whose range is c_{i+1}, and c_{n-1} subsumes c_n. We consider that a class dominates itself.

A contextualized vocabulary matching μ from S into T is *structurally correct* iff, for all $(v_1,e_1,v_2,e_2) \in \mu$ such that v_1 and v_2 are properties:

(i) there is a class f of S such that μ matches f with the domain of v_2 and f dominates e_1 (recall from the definition of vocabulary matching that e_1 is a subclass of the domain of v_1)
(ii) if v_1 is a datatype property, then the range of v_1 is a subtype of the range of v_2
(iii) if v_1 is an object property, then μ matches the range of v_1 with the range of v_2

A concept mapping γ from S into T *induced by* a structurally correct contextualized vocabulary matching μ is a set of rules derived from μ as suggested by the examples in Section 2.2. The rules in γ in turn induce a function $\overline{\gamma}$ that maps sets of triples of S into sets of triples of T.

We say that the declarations of the domain and range of properties, the property characteristics, the cardinality restrictions, and the subclass declarations are the *constraints* of a schema.

We denote the minCardinality and the maxCardinality of a property p by $mC[p]$ and $MC[p]$, respectively. By convention, we take $mC[p]=0$ (and $MC[p]=\infty$), if minCardinality (or maxCardinality) is not declared for p.

A property q is *no less constrained* than a property p iff $mC[p] \leq mC[q]$ and $MC[p] \geq MC[q]$ and, if p is declared as inverse functional, then so is q. Note that this definition applies even if p and q are from different schemas.

Let S and T be two schemas, μ be a structurally correct contextualized vocabulary matching from S into T, and γ be a concept mapping from S into T induced by μ.

Let ρ be a rule in γ of the form $p(x,y) \leftarrow B[x,y]$. By construction, p is a property of T and all classes and properties that occur in $B[x,y]$ belong to S. We introduce a property of S, denoted $prop[B]$, defined by $B[x,y]$. We say that ρ is *correct* iff $prop[B]$ is no less constrained than p. We then say that γ is *correct* iff all rules in γ are correct.

Finally, we say that a constraint α of T *is relevant for* γ iff α uses only concepts that occur in the heads of the rules in γ. We then say that γ is *consistent* iff, if I is a consistent set of triples of S, then the set of triples of T defined by $J = \bar{\gamma}(I)$ satisfies all constraints of T that are relevant for γ.

Lemma 1: Let μ be a structurally correct contextualized vocabulary matching and γ be a concept mapping from S into T induced by μ. Assume that γ is correct. Then, γ is consistent.

(The proof generalizes Examples 2, 3 and 4. See [12] for the details).

3 Instance-Based OWL Schema Matching

In this section, we describe an instance-based process to create contextualized vocabulary matchings that are structurally consistent.

We first recall the matching technique for catalogue schemas based on similarity heuristics introduced in [11]. Briefly, a *catalogue* is a relational database whose schema S has a single table. Given a catalogue state $\boldsymbol{U}_S$, an attribute A of S is represented by the set of values of A that occur in $\boldsymbol{U}_S$, or by the set of pairs (i,v) such that v is the value of A for the object with id i that occurs in $\boldsymbol{U}_S$. If the domain of A is a set of strings, the set of values is replaced by a set of tokens, and the attribute representations are reinterpreted accordingly. Similarity models were then applied to such attribute representations to generate attribute matchings between two catalogue schemas.

We also recall that the instance matching technique of Bilke and Naumann [1] represents each database tuple as a character string and uses k-mean clustering algorithms to find duplicate tuples. However, we note that the representations of the same object in distinct databases may differ in the list of attributes and in the attribute values. As a consequence, we may end up with dissimilar tuples that represent the same object.

Table 2. Example the same book instance representation in eBay and Amazon

eBay	Amazon
isbn-10 = "039577537X"	isbn = "039577537X"
isbn-13 = 9780395775370	ean = 9780395775370
title = "The Tragedy of Romeo and Juliet"	title = "Tragedy of Romeo and Juliet: And Related Readings (Literature Connections)"
author = "William Shakespeare"	author = "William Shakespeare"
publisher = "Houghton Mifflin"	name = "Houghton Mifflin Company"
returnPolicyDetails = "NO RETURNS ARE ACCEPTED"	-
condition = "Like New"	-
binding = "Hardcover"	-
-	listPrice = 18.92
-	currency = "USD"

For example, suppose that we apply the Bilke and Naumann technique to match the two instances that represent the book "The Tragedy of Romeo and Juliet", whose property-value pairs are shown in Table 2. If we measure the similarity between the sets of tokens extracted from all property values of each instance, we obtain a score of 43% of common tokens. By contrast, if we consider only the values of the properties that match, the similarity increases to 70%. However, note that, to improve the instance matching strategy, we used the fact that `am:Book` matches `eb:Book`, and the fact that several properties match.

Combining these observations, we propose the four-step vocabulary matching process outlined as follows:

(1) Generate a preliminary property matching using similarity functions.
(2) Use the property matching obtained in Step (1) to generate: (a) a class matching; and (b) an instance matching.
(3) Use the class matching and the instance matching obtained in Step (2) to generate a refined contextualized property matching.
(4) The final vocabulary matching is the result of the union of the class matching obtained in Step (2) and the property matching obtained in Step (3), adjusted until it becomes structurally correct.

Step (1) generates preliminary property matchings based on the intuition that *"two properties match iff they that have many values in common and few values not in common"*. Step (2) creates class matchings that reflect the intuition that *"two classes match iff they have many matching properties"*. However, to work correctly, Step (2) requires that Step (1) generates preliminary property matchings only for highly similar properties.

For example, in the experiments described in Section 4, with data from the eBay and the Amazon databases, if we use a threshold τ=0.12, then `eb:level` with context `eb:Seller` matches `am:color` with context `am:PCHardware` and `eb:title` with context `eb:Music` matches `am:title` with context `am:Video`. These property matchings may cause classes `eb:Seller` and `am:PCHardware` to match, as well as

`eb:Music` and `am:Video`, depending on the threshold and the total amount of common properties among the classes (as discussed below, class matching depends on the similarity between sets of properties). If we increase the threshold to 0.13, the previous property matchings do not hold, avoiding the above unwanted class matchings.

In what follows, let S and T be two schemas, $\boldsymbol{V}_S$ and $\boldsymbol{V}_T$ be their vocabularies, $\boldsymbol{P}_S$ and $\boldsymbol{P}_T$ be their sets of properties, and $\boldsymbol{C}_S$ and $\boldsymbol{C}_T$ be their sets of classes, respectively. Let $\boldsymbol{U}_S$ and $\boldsymbol{U}_T$ be fixed sets of triples of S and T, respectively, to be used to compute the vocabulary matchings.

Let $\mathcal{U}$ be the universe of all tokens extracted from literals and all URIrefs. Consider a similarity function $\sigma:\mathcal{U}\times\mathcal{U}\rightarrow[0,1]$, a *similarity threshold* $\tau\in[0,1]$ and a *related similarity threshold* $\tau'\in[0,1]$ such that $\tau'<\tau$.

For each property $P\in\boldsymbol{P}_S$, for each class $C\in\boldsymbol{C}_S$ such that C is the domain of P or a subclass of the domain of P, consider the contextualized property $P^C=(P,C)$ and construct the set $o[\boldsymbol{U}_S,P^C]$ of all v such that there are triples of the form (I,P,v) and $(I,\texttt{rdf:type},C')$ in $\boldsymbol{U}_S$, where $C'=C$ or C' is a subclass of C, and likewise for a property in $\boldsymbol{P}_T$. We call $o[\boldsymbol{U}_S,P^C]$ the *observed-value representation* of P^C in $\boldsymbol{U}_S$. This construction explores the fact that P is inherited by all subclasses of its domain.

The *contextualized property matching between S and T induced by* σ *and* τ, and based on the observed-value representation of properties, is the relation μ_P such that

$$(P,C,Q,D)\in\mu_P \text{ iff } \sigma(o[\boldsymbol{U}_S,P^C],o[\boldsymbol{U}_T,Q^D])\geq\tau \tag{11}$$

For each class C in $\boldsymbol{C}_S$, let $props[S,C]$ be the set of properties in $\boldsymbol{P}_S$ whose domain is C or that C inherits from its superclasses, and likewise for classes in $\boldsymbol{C}_T$. We call $props[S,C]$ the *representation* of C in $\boldsymbol{U}_S$.

The *contextualized class matching between S and T induced by* σ, τ and μ_P is the relation $\mu_C\subseteq\boldsymbol{C}_S\times\boldsymbol{C}_T$ such that (recall that $\top$ is the top class)

$$(C,\top,D,\top)\in\mu_C \text{ iff } \sigma(props[S,C],relprops[S,C,T,D]))\geq\tau \tag{12}$$

where $relprops[S,C,T,D]$ denotes the set of properties P of class C of S such that there is a property Q of class D of T such that $(P,C,Q,D)\in\mu_P$. Note that it does not make sense to directly compute $\sigma(props[S,C],props[T,D])$, since $props[S,C]$ and $props[T,D]$ are sets of URIrefs from different vocabularies. To avoid this problem, we replaced $props[T,D]$ by $relprops[S,C,T,D]$.

From the matchings directly induced by σ and τ, the process then derives an instance matching and a refined contextualized property matching, as follows.

Figure 3 shows the algorithm that computes the instance matching. It receives as input S and T, and the class matching μ_C induced by σ, τ and μ_P. It also implicitly receives as input $\boldsymbol{U}_S$ and $\boldsymbol{U}_T$. It outputs an instance matching μ_I between class instances in $\boldsymbol{U}_S$ and $\boldsymbol{U}_T$. In Figure 3, if C is a class in $\boldsymbol{C}_S$, and I is an instance of C in $\boldsymbol{U}_S$, then $t[\boldsymbol{U}_S,C](I)$ denotes the set of tokens extracted from all values v such that, for some property $P\in\boldsymbol{P}_S$, for some property Q in $\boldsymbol{P}_T$, for some class $D\in\boldsymbol{C}_T$, there is a triple (I,P,v) in $\boldsymbol{U}_S$ and there is a quadruple (P,C,Q,D) in μ_P, and likewise for $t[\boldsymbol{U}_T,D](J)$.

Figure 4 shows the algorithm that computes the refined contextualized property matching. It depends on the following additional definitions. For each $(P,C,Q,D)\in\mu_P$ such that $(C,\top,D,\top)\in\mu_C$, construct the set $\boldsymbol{q}$ of triples (I,u,v) such that there are triples of the form (I,P,u) and $(I,\texttt{rdf:type},C)$ in $\boldsymbol{U}_S$, there are triples of the form (J,Q,v) and

```
INSTANCE-MATCHING(S,T,μC)
for each pair of classes (C,D) in S and T
 such that μC matches C with D
 for each pair of instances (I,J) of C and D in US and UT
 if σ(t[US,C](I),t[UT,D](J)) ≥ τ
    then μI = μI ∪ (I,C,J,D)
```

Fig. 3. The class instance matching algorithm

```
CONTEXTUALIZED-PROPERTY-MATCHING(S,T,μC)
for each pair of classes (C,D) in S and T
 such that μC matches C with D
 or C' dominates C and μC matches C' with D
 or μC matches C with D' and D' dominates D
 for each pair (P,Q) of properties of C and D
 X = σ(o[US,PC],o[UT,QD])
 if (C matches D)
 then (s,t)=iv[P,C,Q,D]
 Y = σ(s,t)
 else Y = 0
 if max(X,Y) ≥ τ'
    then μA = μA ∪ (P,C,Q,D)
```

Fig. 4. The contextualized property matching algorithm

(J,`rdf:type`*,D)* in $\boldsymbol{U}_T$, and $(I,C,J,D) \in \mu_I$ (where μ_I is the instance matching of Figure 3). Define $iv[P,C,Q,D]=(\boldsymbol{s},\boldsymbol{t})$ such that $\boldsymbol{s}=\{(I,u)/(\exists v)(I,u,v)\in q\}$ and $\boldsymbol{t}=\{(I,v)/(\exists u)(I,u,v)\in \boldsymbol{q}\}$. We call $\boldsymbol{s}$ the *instance-value representation* of P^C in $\boldsymbol{U}_S$ (and likewise for $\boldsymbol{t}$). This second representation is useful since it helps distinguish properties with similar sets of values, but which refer to distinct instances, matched by μ_I.

Returning to the algorithm in Figure 4, it has the same input as the algorithm in Figure 3, and outputs a contextualized property matching μ_A only between properties whose domains are classes directly or indirectly matched by μ_C. The algorithm uses the maximum of the similarity values computed using the observed-value and the instance-value representations for a pair of properties P and Q, and the more relaxed similarity threshold. Although not shown in Figure 4, object properties receive a special treatment, since their representations are sets of URIrefs that are compared with help of the instance matching μ_I (computed by the algorithm in Figure 3).

The final vocabulary matching μ is the union of the class matching μ_C induced by σ, τ and μ_P and the contextualized property matching μ_A computed by the algorithm in Figure 4. However, μ may have to be adjusted, by dropping matchings, until it becomes structurally correct (details omitted for brevity).

4 Experimental Results

We conducted an experiment to assess the performance of the vocabulary matching process of Section 3, using data about products obtained from Amazon and eBay.

We tested the process with data downloaded from the Web, rather than with the benchmark proposed in Duchateau et al [8], since the benchmark does not include instances and is therefore unsuitable to test our process.

Table 3. Automatically obtained vocabulary matching from eBay into Amazon

#	eBay		Amazon		Match Type
	v1	**e1**	**v2**	**e2**	
1	Books	T	Books	T	tp
2	author	B	author	B	tp
3	edition	B	edition	B	tp
4	format	B	biding	B	tp
5	isbn-10	B	isbn	B	tp
6	isbn-13	B	ean	B	tp
7	editionDesc	B	format	B	*fp*
8	Offer	T	Books		*fp*

We first defined a set of terms, which were used to query the databases. From the query results, we extracted the less frequent terms common to both databases. We then used these terms to once more query the databases. This pre-processing step enhanced the probability of retrieving duplicate objects from the databases, which is essential to evaluate any instance-based schema matching technique. We extracted a total of 116,201 records: 16,410 from Amazon and 99,791 from eBay.

We adopted as similarity functions the contrast model [11], for property matchings, and the cosine distance with TF/IDF, for instance matchings. The experiments lead us to conclude that the contrast model has a better performance when we want to emphasize the difference between two sets of values. This follows because the contrast model has room for calibrating several parameters.

Table 3 shows sample entries of the vocabulary matching obtained. The headings indicate that **e1** is the context of **v1**, and **e2** that of **v2**. Also, "B" abbreviates classes `eb:Book` and `am:Book`.

The rightmost column of Table 3 classifies the matchings: *tp* for true positive, *fp* for false positive and *fn* for false negative. Since the total number (not all shown in Table 3) of true positives is 25, that of false positives is 4 and that of false negatives is 10, the performance measures therefore are:

$$precision = \frac{tp}{tp + fp} = 86\% \quad recall = \frac{tp}{tp + fn} = 71\% \quad fMeasure = 2\frac{precision \cdot recall}{precision + recall} = 78\%$$

Lines 3, 5 and 6 of Table 3 refer to matchings that would have been considered false negatives, if the algorithm in Figure 4 ignored the instance-value representation of properties. In this case, the performance measures would drop to:

$$precision = 82\% \quad recall = 51\% \quad fMeasure = 63\%$$

5 Conclusions

In this paper, we proposed a process to match the vocabularies of pairs of OWL extralite schemas and to create a concept mapping out of a vocabulary matching. The process is instance-based and uses similarity functions to induce vocabulary matchings in a non-trivial way. The last step of the process guarantees that the final vocabulary matching is structurally correct and, therefore, induces a consistent concept mapping. We illustrated the approach with experiments using data available on the Web.

The results described in the paper admit several extensions. In particular, we may extend the process to gradually revise the matchings as new data becomes available, which is typical of a query mediation environment. We may also extend the process to more complex OWL schemas, which requires a strategy to revise the target OWL schema.

Acknowledgements. This work was partly supported by CNPq under grants 142103/2007-1, 301497/2006-0 and 473110/2008-3.

References

1. Bilke, A., Naumann, F.: Schema matching using duplicates. In: Proc. of the 21st Int'l. Conf. on Data Engineering, pp. 69–80
2. Brauner, D.F., Casanova, M.A., Milidiú, R.L.: Towards Gazetteer Integration Through an Instance-based Thesauri Mapping Approach. In: Adv. in Geoinformatics, pp. 235–245. Springer, Heidelberg
3. Brauner, D.F., Gazola, A., Casanova, M.A.: Adaptative matching of database web services export schemas. In: Proc. of the 10th Int'l. Conf. on Enterprise Inf. Systems
4. Brauner, D.F., Intrator, C., Freitas, J.C., Casanova, M.A.: An instance-based approach for matching export schemas of geographical database Web services. In: Proc. of the IX Brazilian Symp. on GeoInformatics (GeoInfo), pp. 109–120
5. Breitman, K., Casanova, M., Truszkowski, W.: Semantic web: concepts, technologies, and applications. Springer, London
6. Casanova, M., Breitman, K., Brauner, D., Marins, A.: Database conceptual schema matching. Computer 40(10), 102–104
7. Castano, S., Ferrara, A., Montanelli, S., Racca, G.: Semantic Information Interoperability in Open Networked Systems. In: Proc. ICSNW, in cooperation with ACM SIGMOD 2004, Paris, France (2004)
8. Duchateau, F., Bellahsène, Z., Hunt, E.: XBenchMatch: a benchmark for XML schema matching tools. In: Proc. 33th Int'l. Conf. on VLDB, Demo Sessions, pp. 1318–1321
9. Euzenat, J., Shvaiko, P.: Ontology matching. Springer, Heidelberg
10. Horrocks, I., Patel-Schneider, P.F., Boley, H., Tabet, S., Grosofand, B., Dean, M.: SWRL: A semantic web rule language combining OWL and RuleML. W3C
11. Leme, L.A.P., Brauner, D.F., Breitman, K.K., Casanova, M.A., Gazola, A.: Matching object catalogues. J. Innovations in Systems and Software Engineering 4(4), 315–328
12. Leme, L.A.P.P.: Conceptual schema matching based on similarity heuristics. D.Sc. Thesis, Dept. Informatics, PUC-Rio

13. Leme, L.A.P.P., et al.: Evaluation of similarity measures and heuristics for simple RDF schema matching. Technical Report 44/08, Dept. Informatics, PUC-Rio
14. Quine, W.V.: Ontological Relativity. J. of Philosophy 65(7), 185–212
15. Rahm, E., Bernstein, P.: A survey of approaches to automatic schema matching. The VLDB Journal 10(4), 334–350
16. Wang, J., Wen, J., Lochovsky, F., Ma, W.: Instance-based schema matching for web databases by domain-specific query probing. In: Proc. 13th Int'l. Conf. on VLDB, pp. 408–419.

The Integrative Role of IT in Product and Process Innovation: Growth and Productivity Outcomes for Manufacturing

Louis Raymond[1], Anne-Marie Croteau[2], and François Bergeron[3]

[1] Institut de recherche sur les PME, Université du Québec à Trois-Rivières
Trois-Rivières, Canada
[2] John Molson School of Business, Concordia University, Montreal, Canada
[3] Télé-université, Université du Québec à Montréal, Québec, Canada

Abstract. The assimilation of IT for business process integration plays an integrative role by providing an organization with the ability to exploit innovation opportunities with the purpose of increasing their growth and productivity. Based on survey data obtained from 309 Canadian manufacturing SMEs, this study aims at a deeper understanding of the assimilation of IT for business process integration with regard to product and process innovation. The first objective is to identify the effect of the assimilation of IT for business process integration on growth and productivity. The second objective is to verify if the assimilation of IT for business process integration varies amongst low, medium and high-tech SMEs. Results indicate that the assimilation of IT for business process integration depends upon the type of innovation. It also varies as per the technological intensity of the firms. The assimilation of IT for business process integration has two effects: it increases the growth of manufacturing SMEs by enabling product innovation; but it decreases their productivity by impeding the process innovation.

Keywords: IT assimilation, Integration, Growth, Productivity, SME, Innovation, R&D, Manufacturing.

1 Introduction

Innovation has long been considered as the key factor for the survival, growth and development of small and medium-sized enterprises (SMEs) [1,2]. For these organizations, a greater innovation capacity is deemed to counterbalance their greater vulnerability in a globalized business environment and in an economy that is now knowledge-based [3,4]. Innovation is defined as "the economic application of a new idea" [5, p. 270]. It encompasses two components: product and process innovation, where product innovation refers to a new or modified version of a product; and process innovation looks into a new or modified way of making a product [5].

In response to increased competitive pressures brought about by globalization, the manufacturing strategy of SMEs in the last decade has been implemented in good part through the adoption and assimilation of IT in the form of planning and logistics

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 27–39, 2009.
© Springer-Verlag Berlin Heidelberg 2009

applications such as ERP and EDI [6], primarily designed to integrate cross-functional and inter-organizational business processes [7,8,9]. But while information technologies are deemed to enable manufacturing SMEs to grow and be more productive by creating business value in synergy with other organizational factors [10], their specific role with regard to product and process innovation needs further investigation. In theory, the assimilation of IT for the integration of business processes is deemed to provide an organization with the "ability to accomplish speed, accuracy, and cost economy in the exploitation of innovation opportunities" [11, p. 246].

The present study aims at a deeper understanding of the role played by IT with regard to product and process innovation. The first objective of this research is to identify the enabling (and/or disabling) effect of IT upon innovation in manufacturing SMEs, that is, in terms of growth and productivity. The second objective is to verify if this effect is subject to industry influences, given that mechanisms such as investments in R&D constitute an "innovation system" in a given industry or sector [12]. Therefore, the research question is then formulated as follows: To what extent does IT have an enabling effect with regard to innovation in manufacturing SMEs?

2 Assimilation of IT for Business Process Integration

In a business environment that is becoming more and more complex, manufacturing SMEs may act strategically in two basic ways. Growth-oriented firms increase their competitiveness by seeking new markets and putting the emphasis on technological leadership and product innovation [13]. Other manufacturing SMEs, more defensive in their outlook, focus on productivity in terms of reduced costs and improved delivery capabilities, by increasing the flexibility of their productive apparatus and emphasizing process innovation [14].

Hence, product innovation allows SMEs to improve or maintain their position in the market and their relationship with customers, and thus grow, while process innovation aims to improve their productivity by reducing production costs and increasing their operational agility, thus becoming more competitive [15]. Also, best product development practices such as concurrent engineering are founded on the coordination and integration of both product innovation and process innovation [16].

In empirical studies of innovation in SMEs, researchers have sought to explain why certain firms innovate more successfully than others by identifying certain strategic capabilities as "critical success factors" of innovation [17], including technological integration capabilities in particular [18]. A review of empirical studies in the manufacturing sector reveals that 43% of SMEs aimed at both product and process innovation, 37% aimed at product innovation solely, and only 1% at process innovation exclusively [2]. Although there is a need to further investigate process innovation specifically, Martinez-Ros [19] have found that product and process innovation are interdependent and closely linked. Thus, as recommended by Becheikh, Landry and Amara [2], both product and process must be distinctively factored into innovation.

Innovation and the ensuing competitive advantage would derive from leveraging complementary strategic capabilities, most notably R&D capabilities, networking capabilities and technological capabilities. For instance, many small firms compensate

for their lack of internal means and competencies to engage in R&D by cooperating with other firms in the area of technology and innovation [20], and by relying on business partners such as customers, suppliers and research centers as a source of innovation [21].

A number of manufacturing SMEs also assimilate advanced manufacturing technologies such as computer-aided design and manufacturing (CAD/CAM) and flexible manufacturing systems (FMS) that enable them to achieve a competitive advantage with more flexibility, reduced delay (from product design to introduction on the market) and quick response to market changes [22].

3 Research Model and Hypotheses

As presented in Figure 1, the research model hypothesizes that the effect of product and process innovation upon the firm's growth and productivity will be respectively enabled and disabled by its assimilation of IT for business process integration, that is, by its use of applications such as MRP-II, ERP and EDI whose ultimate aim resides in the "seamless" integration of business processes across functions and across organizations [23].

Product innovation, be it incremental or fundamental [24], implies the introduction of a new product that maintains or increases a market share which translates into growth [5]. Process innovation is known to lead to improved productivity [25]. Because both product and process innovation are closely interrelated both should positively factor into innovation which should contribute to an increase in growth and productivity. Therefore the first hypothesis is the following:

Hypothesis 1a: There is a positive relationship between innovation and growth.

Hypothesis 1b: There is a positive relationship between innovation and productivity.

The role of IT for business process integration is related to standardization of the core business processes within a firm and / or with its business partners [8,26]. However, the implementation of integrative IT does not always translate into a true integration [27]. Complete integration normally increases the visibility of the information but also the flexibility in accessing it [28]. However this does not happen easily; in fact it may turn out to be dysfunctional unless the organization reaches a high level of agility [26,28].

Implementing integrative IT such as ERP helps most firms to improve the synchronization of data and systems amongst their suppliers, customers and partners. Those efforts are translated into an increased level of access to the information which allows them to respond better and quicker to the market and therefore increase their growth [29].

Hypothesis 2: The greater the firm's assimilation of IT for business process integration, the greater the impact of product and process innovation on its growth.

Business process integration is a characteristic of manufacturing organizations that bears both an opposing and a complementary relationship to manufacturing flexibility or operational agility. On one hand, integrated processes allow for greater sharing

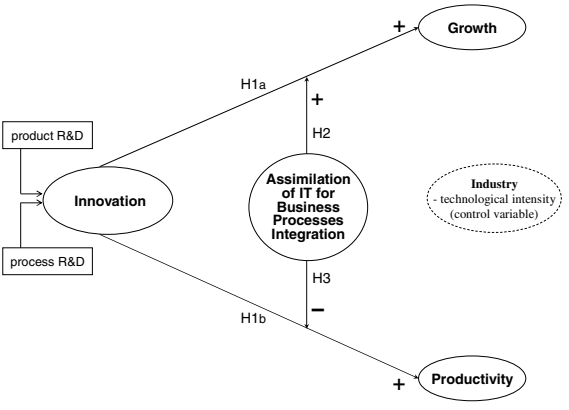


Fig. 1. Research Model

of new information, thus insuring quicker response to changes in the environment and increasing the organization's flexibility. On the other hand, the more an organization is integrated, the harder it is to "disconnect" itself [30]. IT for business process integration such as ERP has thus been qualified as "rigid" rather than "malleable" technology [31]. It has also been found that the more firms adopt integrated technologies, the less flexible they are [32].

Hypothesis 3: The greater the firm's assimilation of IT for business process integration, the lesser the impact of product and process innovation on its productivity.

Note that these hypotheses imply a "fit as moderation" alignment perspective [33], wherein fit is conceptualized as the interaction between IT and innovation. Thus, following Bharadwaj, Bharadwaj and Konsynski's [17] seminal IT alignment research proposition, IT for business process integration is hypothesized to moderate the relationship between the SME's strategic capabilities, in terms of innovation, and its organizational performance, in terms of growth and productivity.

Innovation is susceptible to industry effects, as observed in many studies that have demonstrated the influence of the industrial sector's technological intensity, growth, and structure [2]. For instance, product innovation is deemed to be stronger in sectors of higher technological intensity such as electronics and biotechnology [5]. Also, prior research has confirmed the theoretical and empirical importance of industry as a contingency factor in the relationship between innovation and organizational performance [34,35]. It is thus important to be able to distinguish between firm and industry effects when testing the research hypotheses [36], which is why the research model includes the technological intensity of the industrial sector as a control variable.

4 Research Method

4.1 Data Collection

The research data were obtained from a database created by a university research center, containing information on 309 Canadian manufacturing SMEs. With the

collaboration of an industry association to which most of these firms belong, the database was created by having the SMEs' chief executive and functional executives such as the controller, human resources manager, and production manager fill out a questionnaire to provide data on the practices and results of their firm and add their firm's financial statements for the last five years. Anonymity and confidentiality is preserved by having the questionnaires transit through the industry association so that firms are known by the research center only by an alphanumeric identifier assigned by the association. Once all the questionnaire data and financial statements have been manually verified by the research center's personnel, they are typed in via validation software and entered in the database as valid data, ready for benchmarking. In exchange for these data, the firms are provided with a complete comparative diagnostic of their overall situation in terms of performance and vulnerability (further information on the diagnosis system and on data collection and validation can be found in St-Pierre and Delisle [37].

4.2 Measurement

Based upon the effect of R&D investments on the subsequent growth of the firm, as confirmed in the literature [38], these investments can be used as an indicator of the SME's capacity or propensity to innovate [39,40], and particularly in the context of SMEs [De Jong and Vermeulen 2007]. Investment in R&D is in fact one of the most important mechanisms that constitute the "innovation system" in a given sector or industry [12]. Innovation is thus measured in this study by product R&D and process R&D as surrogate indicators. In line with common measurement practice with regard to R&D and innovation [15], the intensity of product and process R&D activities is measured by two ratios, namely product R&D budget over number of employees and process R&D budget over number of employees. Following Brandyberry, Rai and White [32], the assimilation of IT for business process integration is measured by asking the operations manager to evaluate the extent to which advanced manufacturing applications implemented are actually integrated within the organization, on a scale of 1 (low) to 5 (high). By summing these evaluations over six "planning and logistics" applications, using Kotha and Swamidass' [42] categorization of advanced manufacturing technology, one thus obtains a score (ranging from 0 to 30) of the assimilation by the firm of IT for business process integration. The most widely-used productivity indicator was selected, directly related to the firm's manufacturing systems, that is, the productivity of the workforce as measured by the gross profit per employee. The indicator of growth is also one that is most commonly used, that is, the average growth in sales over the last three years.

4.3 Sample

For the study's purposes, a manufacturing SME is defined as an enterprise with 20 or more employees and less than 500, corresponding to the lower bound used by the European Union [34] and the upper bound used in North American research [43]. The size of the sampled firms thus varies between 20 and 405 employees, with a median of 49, whereas annual sales vary from 0.4 to 55 million Canadian dollars, with a median of 6. More than fifteen industrial sectors are represented, including metal products

(27.5% of the sampled firms), wood (14%), plastics and rubber (13%), electrical products (6.5%), food and beverage (6%), and machinery (5.5%). Being relatively representative of Canadian manufacturing SMEs with regard to size and industry, 104 of the sampled firms (34%) operate in a sector whose technological level is low, 153 (49%) in a medium to low-tech sector, and 52 (17%) in a medium to high-tech sector, there being no high-tech firms based on the OECD classification [44].

5 Results

As shown in Table 1, the first descriptive results pertain to the levels of IT adoption and assimilation in manufacturing SMEs, including manufacturing planning and logistics applications such as computer-based production scheduling, bar-coding, EDI, MRP, MRP-II and ERP that aim to integrate business processes and thus constitute "plant information systems" [7]. It seems that it is still a minority of SMEs that have adopted IT for purposes of integration, including EDI (22% adoption rate), MRP-II (10%) and ERP (9%). One could surmise that the sampled SMEs, in responding to the challenges of globalization, would be oriented more on manufacturing flexibility or operational agility than on integration.

Table 1. Levels of adoption and assimilation of IT for business process integration

Logistics/Planning applications (n = 309) [IT for business process integration]	Adoption rate	Assimilation[a]
Computer-based production scheduling	37 %	3.3
Computer-based bar-coding	29 %	3.7
Electronic data interchange (EDI)	22 %	3.5
Materials requirement planning (MRP)	20 %	3.1
Manufacturing resource planning (MRP-II)	10 %	2.8
Enterprise resource planning (ERP)	9 %	3.3

[a]Perceived mastery of the technology or application adopted (low : 1, 2, 3, 4, 5 : high).

The descriptive statistics of the research variables are presented in Table 2, the mean being broken down by the technological intensity of the firms. SMEs in medium to high-tech sectors show the highest levels of product innovation and productivity, while their level of process innovation is equal to those in the low to medium-tech sectors. Note also that 22% of the variance in product innovation is explained by industry effects rather than by firm effects, whereas there are no industry effects with regard to the assimilation of IT.

5.1 Estimation of Model Parameters

Structural equation modeling was used to test the relationships proposed in the research model. The PLS method was used in view of its capacity to correctly estimate interaction effects [45]. These effects were obtained by using the product of the variables measuring the innovation and IT for business process integration constructs to form an interaction construct. The potential influence of industry on the

Table 2. Descriptive statistics and breakdown of the research variables by industry

Industry[a] / Variable	All SMEs (n = 309) mean s.d. min max	low-tech SMEs (n = 104) mean	low to med.-tech (n = 153) mean	medium to high-tech (n = 52) mean	Anova F	% of variance explained by Industry
Growth[b]	0.17 0.23 -0.29 1.85	0.17	0.17	0.18	0.1	0 %
Productivity[c]	47022 45651 -3641 90261	39173_2	$44857_{2,1}$	69089_1	8.1***	5 %
Innovation-product R&D[d]	1155 2805 0 26800	302_3	768_2	4001_1	41.8***	22 %
Innovation-process R&D[e]	381 681 0 5714	192_2	482_1	462_1	6.3***	4 %
Assimilation of IT for bus. process integration[f]	7.0 5.7 0.0 28	6.7	7.1	7.1	0.2	0 %

***: $p < 0.001$

$_{1,2,3}$*Nota*. Within rows, different subscripts indicate significant ($p < .05$) pairwise differences between means on Tamhane's T2 test.

[a]technological intensity associated to the industrial sector following the OECD's (2005b) classification.
- low-tech: wood, food and beverage, furniture, clothing, textile, printing, paper, leather and others.
- low to medium-tech: metal products and transformation, rubber and plastics, mining products, construction, mineral products, others.
- medium to high-tech: electrical products, machinery, chemical products, transportation equipment and others

[b]average growth in net sales over the last 3 years.

[c]gross profit per employee = (gross profit) / no. of production employees.

[d]product R&D budget / no. of employees.

[e]process R&D budget / no. of employees.

[f]$\Sigma_{k=1,6}$[assimilation of application$_k$].

results were estimated by testing the model anew for each of three sub-samples, that is, for the SMEs operating in industrial sectors of low, medium-low and medium-high technological intensity respectively.

Given that all constructs in the research model are formative, the first structural model results, presented in Table 3, are in regard to the variables' weight upon their associated construct (measurement model), as estimated by PLS. Note also that there is no multicollinearity as the two independent constructs, Innovation and IT for

Table 3. Variables' weight upon their associated construct as estimated by PLS

	All SMEs (n = 309)	Low-tech SMEs (n = 104)	Low to med-tech SMEs (n = 153)	Med. to high-tech SMEs (n = 52)
	weight	weight	weight	weight
Innovation				
product R&D	0.76	1.01	0.99	0.89
process R&D	0.57	-0.27	-0.29	0.38
Innov. x IT for business process integration[a]	0.00	-0.59	0.42	0.00
IT x product R&D	1.00	1.14	0.87	1.00
IT x process R&D				
Growth				
average sales growth for last 3 years	1.00	1.00	1.00	1.00
Productivity				
gross profit per employee	1.00	1.00	1.00	1.00

[a]interaction construct.

business process integration x Innovation, are uncorrelated (R = -0.08), as are the two performance constructs, Growth and Productivity (R = -0.02).

5.2 Test of Research Hypotheses

The two research hypotheses are tested by assessing the direction, strength and level of significance of the path coefficients estimated by PLS, as presented in Table 4. This research investigates the effect of IT for business process integration on the relationship between SME's process and product innovation, and organizational performance measured in terms of growth and productivity. Overall, the main results indicate that innovation has a direct positive effect on both growth and productivity, while IT interaction has a positive effect on growth but a negative one on productivity. Although innovation leads to growth and productivity, the level of IT for business process integration in the firms plays a different role dependent upon the performance objective. In terms of growth, IT for business process integration is associated with a positive effect on growth, in that a higher level of IT for business process integration is associated with higher organizational growth. IT for business process integration is therefore beneficial to SME innovation in that respect. However, the opposite is observed for productivity.

The results indicate that IT for business process integration is associated with a negative effect on productivity; organizations that innovate in a more integrated IT environment have a lower productivity than organizations that innovate in a less integrated IT environment.

Table 4. Results of testing the research model (path coefficients as estimated by PLS)

	All SMEs (n = 309)		Low-tech SMEs (n = 104)		Low to med.-tech SMEs (n =153)		Med. to high-tech SMEs (n = 52)	
	Growth	Product.	Growth	Product.	Growth	Product.	Growth	Product.
Innovation	0.174**	0.201***	-0.144[a]	0.359*	-0.044	0.215*	0.482[a]	0.026
Innov. x IT for Integr.	0.126**	-0.278***	0.169	-0.226*	0.115*	-0.120*	0.140	-0.657***
R^2 (%)	4.2	12.9	4.8	17.5	1.5	6.1	21.6	44.2

[a]$p < 0.1$ *: $p < 0.05$ **: $p < 0.01$ ***: $p < 0.001$
Nota. Significance levels were obtained by bootstrapping.

These relationships vary however, depending upon the technological intensity of the SMEs as is shown in Table 4. The main relationships are as follows: The relationship between innovation and growth changes from slightly negative (γ = -0.144) in the case of low-tech SMEs to strongly positive (γ = 0.482) for high-tech SMEs. The opposite is true for productivity, where the relationship between innovation and productivity changes from strongly positive (γ = 0.359) for the low-tech SME to a non-significant relationship (γ = 0.026) for the high-tech SME.

The innovation construct is based on product and process R&D innovation, and the observed direct relationships with growth and productivity can be interpreted with these two kinds of innovation process in mind. However, all the conclusions concerning the interaction between IT for business process integration, growth and productivity concern only process innovation, since the specific measurement of the interaction between IT for business process integration and product innovation R&D did not weigh in sufficiently in the structural model (see Table 3). Therefore, the IT for business process integration interaction effect with productivity and growth specifically concerns process innovation only. The fact that process innovation is the only significant factor is in conformity with Utterback's (1994) revised model of the innovation life cycle. In this model, process innovation efforts are deemed to occur earlier and have greater effect whereas product innovation efforts are deemed to phase out early and have less effect, due to the enabling role of IT.

6 Discussion and Conclusions

Several interpretations of the results can be made. Overall, IT for business process integration shows a positive relationship with growth and a negative one with productivity. This might be due to the fact that a highly integrated firm hampers the

possibility to increase productivity where the proposed changes in the processes might conflict with actual processes. The human and technical problems as well as the time needed to introduce the new processes directly affect the gross margin per employee. The more the actual processes are integrated, the less it is possible to change them without decreasing productivity, at least in the short term. However, this conflict does not show up in the relationship with growth. It might be that highly integrated processes allow the firm to rapidly introduce new products on the market. This is observed overall ($\gamma = 0.126$) and specifically for medium-to-low tech SMEs ($\gamma = 0.115$). IT for business process integration includes both internal and external integration. Thus, the time needed to launch a new product resulting from product R&D innovation can be shortened significantly if the internal processes are highly integrated with the external processes, i.e. the backbone of the extended value chain. In this case, organizational growth, measured in terms of increased sales, show positive improvements.

The target period of the measurements may bring another explanation. While IT for business process integration seems a legitimate goal, it might not be profitable at least in the short run. In the long run, adjustments can likely be made where new processes are implemented and streamlined for a greater organizational productivity.

This study has certain limitations that must be mentioned. Given that the sample is composed of firms that have chosen to undertake an organizational diagnostic exercise, there could thus be a sample bias. These firms may differ from the general population in regard to their innovativeness, assimilation of IT for business process integration, and performance [47]. Other than the nature of the sample, another limit associated to survey research pertains to the use of a perceptual measure of IT assimilation that demands prudence in generalizing results. The cross-sectional rather than longitudinal nature of the study moreover implies that the results do not necessarily reflect the long-term enabling effects of IT on innovation.

One can conclude from the results of this study that IT "does matter" for innovation in manufacturing SMEs. IT matters in different ways however, depending upon the firm's innovation strategy. This aspect of the firm's competitive strategy may be outward-bound and growth-oriented, say for the "prospector" type of SME as defined in Miles and Snow's [48] strategic typology, or it may be inward-bound and productivity-oriented, say for the "defender" type. While the assimilation of IT for business process integration is seen to enable product innovation by increasing the growth of manufacturing SMEs, it tends to disable process innovation by decreasing the productivity of these organizations. The integrative role of IT in manufacturing is also shown here to vary across industries, and thus the need for future research to take industry effects into account. Returning anew to the "productivity paradox", IT for business process integration such as ERP systems can indeed be counter-productive, and "seamless integration" can induce rigidities that run counter to process innovation aims [49]. Further understanding of the potential dialogic between IT for business process integration and IT for flexibility is needed if these technologies are to effectively enable the operational and managerial processes of SMEs, thus improving the organizational performance of these firms and helping them achieve "world-class" manufacturing status.

References

1. Acs, Z.J., Audretsch, D.B.: Innovation and Small Firms. MIT Press, Cambridge (1990)
2. Becheikh, N., Landry, R., Amara, N.: Lessons from innovation empirical studies in the manufacturing sector: A systematic review of the literature from 1993-2003. Technovation 26(5/6), 644–664 (2006)
3. Hoffman, K., Parejo, M., Bessant, J., Perren, L.: Small firms, R&D, technology and innovation in the UK: a literature review. Technovation 18(1), 39–55 (1998)
4. Roper, S., Love, J.H.: Product innovation and small business growth: A comparison of the strategies of German, U.K. and Irish companies. Research Policy 31, 1087–1102 (2002)
5. Subrahmanya, M.H.B.: Pattern of technological innovations in small enterprises: a comparative perspective of Bangalore (India) and Northeast England (UK). Technovation 25, 269–280 (2005)
6. Muscatello, J.R., Small, M.H., Chen, I.J.: Implementing enterprise resource planning (ERP) systems in small and midsize manufacturing firms. International Journal of Operations & Production Management 23(8), 850–871 (2003)
7. Banker, R., Bardhan, I., Chang, H., Lin, S.: Impact of manufacturing practices on adoption of plant information systems. In: Proceedings of the Twenty-Fourth International Conference on Information Systems, pp. 233–245 (2003)
8. Barki, H., Pinsonneault, A.: A model of organizational integration, implementation effort, and performance. Organization Science 16(2), 165–179 (2005)
9. Park, K., Kusiak, A.: Enterprise resource planning (ERP) operations support system for maintaining process integration. International Journal of Production Research 43(19), 3959–3982 (2005)
10. Kohli, R., Grover, V.: Business value of IT: An essay on expanding research directions to keep up with the times. Journal of the Association for Information Systems 9(1), 23–39 (2008)
11. Sambamurthy, V., Bharadwaj, A., Grover, V.: Shaping agility through digital options: Reconceptualizing the role of information technology in contemporary firms. MIS Quarterly 27(2), 237–263 (2003)
12. Baldwin, J.R., Hanel, P.: Innovation and Knowledge Creation in an Open Economy: Canadian Industry and International Implications. Cambridge University Press, Cambridge (2003)
13. Özsomer, A., Calantone, R.J., Di Benedetto, A.: What makes firms more innovative? A look at organizational and environmental factors. Journal of Business & Industrial Marketing 12(6), 400–416 (1997)
14. Sum, C., Kow, L.S.-J., Chen, C.-S.: A taxonomy of operations strategies of high performing small and medium enterprises in Singapore. International Journal of Operations & Production Management 24(3), 321–345 (2004)
15. OECD, Oslo Manual: Guidelines for Collecting and Interpreting Innovation Data, 3rd edn., OECD, Paris (2005a)
16. Lim, L.P.L., Garnsey, E., Gregory, M.: Product and process innovation in biopharmaceuticals: a new perspective on development. R&D Management 36(1), 27–36 (2006)
17. Bharadwaj, A.S., Bharadwaj, S.G., Konsynski, B.R.: The moderator role of information technology in firm performance: A conceptual model and research propositions. In: Proceedings of the Sixteenth International Conference on Information Systems, pp. 183–188 (1995)

18. Swink, M., Nair, A.: Capturing the competitive advantage of AMT: Design-Manufacturing integration as a complementary asset. Journal of Operations Management 25(3), 736–754 (2007)
19. Martinez-Ros, E.: Explaining the decisions to carry out product and process innovations: the Spanish case. Journal of High Technology Management Research 10(2), 223–242 (1999)
20. Lindman, M.T.: Open or closed strategy in developing new products? A case study of industrial NPD in SMEs. European Journal of Innovation Management 5(4), 224–236 (2002)
21. Avermaete, T., Viaene, J., Morgan, E.J., Crawford, N.: Determinants of innovation in small food firms. European Journal of Innovation Management 6(1), 8–17 (2003)
22. Ariss, S.S., Raghunathan, T.S., Kunnathar, A.: Factors affecting the adoption of advanced manufacturing technology in small firms. SAM Advanced Management Journal 56(2), 14–21 (2000)
23. Markus, M.L.: Reflections on the systems integration enterprise. Business Process Management Journal 7(3), 171–180 (2001)
24. Fergurson, P.R., Fergurson, G.J.: Industrial Economics: Issues and Perspectives, 2nd edn., Palgrave, Hampshire (1994)
25. Heygate, R.: Why are we bungling process innovation? The McKinsey Quarterly 2, 130–141 (1996)
26. Ross, J.W.: Creating a strategic IT architecture competency: learning in stages. MIS Quarterly Executive 2(1), 31–43 (2003)
27. Bagchi, P.K., Skjoett-Larsen, T.: Integration of information technology and organizations in a supply chain. The International Journal of Logistics Management 14(1), 89–108 (2002)
28. Evgeniou, T.: Information integration and information strategies for adaptive enterprises. European Management Journal 20(5), 486–494 (2002)
29. Lee, H., Farhoomand, A., Ho, P.: Innovation through supply chain recognition. MIS Quarterly Executive 3(3), 131–142 (2004)
30. Markus, M.L.: Paradigm shifts: e-business and business systems integration. Communications of the Association for Information Systems 4, article 10, 1–44 (2000)
31. Elbanna, A.M.: The validity of the improvisation argument in the implementation of rigid technology: the case of ERP systems. Journal of Information Technology 21, 165–175 (2006)
32. Brandyberry, A., Rai, A., White, G.P.: Intermediate performance impacts of advanced manufacturing technology systems: an empirical investigation. Decision Sciences 30(4), 993–1020 (1999)
33. Venkatraman, N.: The concept of fit in strategy research: toward verbal and statistical correspondence. Academy of Management Review 14(3), 423–444 (1989)
34. Kalantaridis, C., Pheby, J.: Processes of innovation among manufacturing SMEs: the experience of Bedfordshire. Entrepreneurship & Regional Development 11, 57–78 (1999)
35. Tidd, J., Bessant, J., Pavitt, K.: Managing Innovation: Integrating Technological, Market and Organizational Change, 3rd edn. John Wiley, Chichester (2005)
36. Mauri, A.J., Michaels, M.P.: Firm and industry effects within strategic management: An empirical examination. Strategic Management Journal 19(3), 211–219 (1998)
37. St-Pierre, J., Delisle, S.: An expert diagnosis system for the benchmarking of SMEs' performance. Benchmarking: An International Journal 13(1/2), 106–119 (2006)

38. Co, H.C., Chew, K.S.: Performance and R&D expenditures in American and Japanese manufacturing firms. International Journal of Production Research 35(12), 3333–3348 (1997)
39. Qian, G., Li, L.: Profitability of small and medium-sized enterprises in high-tech industries: The case of the biotechnology industry. Strategic Management Journal 24(9), 881–887 (2003)
40. Wolff, J.A., Pett, T.L.: Small-firm performance: modeling the role of product and process improvements. Journal of Small Business Management 44(2), 268–284 (2006)
41. De Jong, J.P.J., Vermeulen, P.A.M.: Determinants of product innovation in small firms. International Small Business Journal 24(6), 587–609 (2007)
42. Elbanna, A.R.: The validity of the improvisation argument in the implementation of rigid technology: the case of ERP systems. Journal of Information Technology 21(3), 165–175 (2006)
43. Kotha, S., Swamidass, P.M.: Strategy, advanced manufacturing technology and performance: empirical evidence from U.S. manufacturing firms. Journal of Operations Management 18(3), 257–277 (2000)
44. Mittelstaedt, J.D., Harben, G.N., Ward, W.A.: How small is too small? Firm size as a barrier to exporting from the United States. Journal of Small Business Management 41(1), 68–84 (2003)
45. OECD, OECD Science, Technology and Industry Scoreboard 2005, OECD, Paris (2005b), http://puck.sourceoecd.org/vl=380292/cl=28/nw=1/rpsv/scoreboard/index.htm
46. Chin, W.W., Marcolin, B.L., Newsted, P.R.: A partial least squares latent variable modeling approach for measuring interaction effects: results from a Monte Carlo simulation study and voice mail emotion/adoption study. In: Proceedings of the Seventeenth International Conference on Information Systems, pp. 21–41 (1996)
47. Utterback, J.M.: Mastering the dynamics of innovation. Harvard Business School Press, Boston, Massachusetts (1994)
48. Cassell, C., Nadin, S., Gray, M.O.: The use and effectiveness of benchmarking in SMEs. Benchmarking: An International Journal 8(3), 212–222 (2001)
49. Miles, R.E., Snow, C.C.: Organizational Strategy, Structure, and Process. McGraw-Hill, New York (1978)
50. Raymond, L.: Operations Management and Advanced Manufacturing Technologies in SMEs: A Contingency Approach. Journal of Manufacturing Technology Management 16(8), 936–955 (2005)

Vectorizing Instance-Based Integration Processes

Matthias Boehm[1], Dirk Habich[2], Steffen Preissler[2],
Wolfgang Lehner[2], and Uwe Wloka[1]

[1] Dresden University of Applied Sciences, Database Group
mboehm@informatik.htw-dresden.de,
wloka@informatik.htw-dresden.de
[2] Dresden University of Technology, Database Technology Group
dirk.habich@tu-dresden.de,
steffen.preissler@tu-dresden.de,
wolfgang.lehner@tu-dresden.de

Abstract. The inefficiency of integration processes—as an abstraction of workflow-based integration tasks—is often reasoned by low resource utilization and significant waiting times for external systems. Due to the increasing use of integration processes within IT infrastructures, the throughput optimization has high influence on the overall performance of such an infrastructure. In the area of computational engineering, low resource utilization is addressed with vectorization techniques. In this paper, we introduce the concept of vectorization in the context of integration processes in order to achieve a higher degree of parallelism. Here, transactional behavior and serialized execution must be ensured. In conclusion of our evaluation, the message throughput can be significantly increased.

Keywords: Vectorization, Integration processes, Throughput optimization, Pipes and filters, Instance-based.

1 Introduction

Integration processes—as an abstraction of workflow-based integration tasks—are typically executed with the *instance-based execution model*. This implies that incoming messages are serialized in incoming order, and this order is then used to execute single-threaded instances of process plans. Example system categories for that execution model are EAI (Enterprise Application Integration) servers, WfMS (Workflow Management Systems) and WSMS (Web Service Management Systems). Workflow-based integration platforms usually do not reach high resource utilization because of (1) the existence of single-threaded process instances in parallel processor architectures, (2) significant waiting times for external systems, and (3) IO bottlenecks (message persistence for recovery processing). Hence, the throughput—in the sense of processed integration process plan instances per time period—is not optimal and can be significantly optimized using a higher degree of parallelism. The opposite to the *instance-based execution model* is the *pipes and filters execution model*. Here, each operator is conceptually a single thread, and each edge between two operators contains a message queue. Hence, a high degree of parallelism is reached. This is typical for DSMS (Data Stream Management Systems) and ETL (Extraction Transformation Loading) tools.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 40–52, 2009.
© Springer-Verlag Berlin Heidelberg 2009

Our approach is to introduce the vectorization of integration processes as an internal optimization concept in order to increase the throughput of integration platforms. We use the term vectorization in the sense of a transformation from the instance-based to the pipes-and-filters execution model. Note that this is an analogy to computational engineering, where vectorization is classified (according to Flynn) as SIMD (single instruction, multiple data) or in special cases as MIMD (multiple instruction, multiple data). We use this analogy because in the pipes-and-filters execution model, sequences (vectors) of messages are executed by a single operator. Here, specific constraints like the serialization of external behavior and the transactional behavior (recoverability) must be ensured. Finally, there is the need for execution model transparency. Thus, the user should think of an *instance-based execution model* as the used logical model.

In order to overcome the problem of low message throughput (caused by low resource utilization), we make the following contributions:

- In Section 2, we explain requirements for integration processes and we formally define the integration process vectorization problem.
- Subsequently, in Section 3, we introduce our novel approach for process plan rewriting in order to apply the vectorization of process instances.
- Based on those details, we present selected results of our exhaustive experimental evaluation in Section 4.

Finally, we analyze related work in Section 5 and conclude in Section 6.

2 Problem Description

In this section, we emphasize the assumptions and requirements that lead to our idea of throughput optimization. Here, we formally define the integration process vectorization problem, survey possible application areas, and finally give a solution overview.

2.1 Assumptions and Requirements

Figure 1 illustrates a generalized integration platform architecture for instance-based integration processes. Here, the key characteristics are a set of inbound adapters (passive listeners), several message queues, a central process engine, and a set of outbound adapters (active services). The message queues are used as logical serialization elements within the asynchronous execution model. However, the synchronous as well as the asynchronous execution of process plans is supported. Further, the process engine

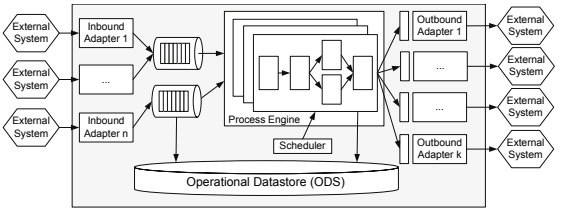


Fig. 1. Integration Platform Architecture

is instance-based, which means that for each subsequent message in a queue, a new instance (one thread) of the specified process plan is created and executed serially.

In the context of integration processes, the throughput maximization rather than the execution time minimization is the major optimization objective. Further, we assume that integration platforms typically do not have a 100-percent resource utilization. This is mainly caused by (1) significant waiting times for external system invocations, (2) the trend towards multi-processor architectures, and (3) the IO bottleneck due to the need for message persistence for recoverability issues. Hence, by increasing the degree of parallelism, the message throughput can be significantly improved.

Due to the need for logical serialization of process plan instances, simple multi-threading of single instances is not applicable. As presented in [1], we must ensure that messages do not outrun other messages; for this purpose, we use logical serialization concepts such as message queues.

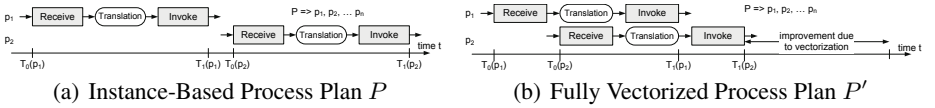
Example 1. Message Outrun Anomaly: *Assume two message types: orders, M_O, and customer, M_C. Messages of those different types are executed by different integration processes P_O and P_C with $M_O \rightarrow P_O$ and $M_C \rightarrow P_C$. Both process types comprise the receipt of a message, the schema mapping and the invocation of an external system s_1. Further, assume that the customer master data must be propagated to the external system s_1 before the customer's first order can be processed. In addition to that, the inventory is maintained during order processing. In the serialized case, messages of both types are serialized. Hence, they cannot outrun each other. In the non-serialized case, an order message can outrun the corresponding customer information. This might result in a referential integrity conflict within the target system s_1.*

However, the serialized execution of process instances is not always required. We can weaken this to serialized external behavior of process plan instances.

2.2 Optimization Problem

Now, we formally define the integration process vectorization problem. Figure 2(a) illustrates the temporal aspects of a typical instance-based integration process. Here, a message is received from a message queue (`Receive`), then a schema mapping (`Translation`) is processed and finally, the message is sent to an external system (`Invoke`). In this case, different instances of this process plan are executed in serialized order. In contrast to this, Figure 2(b) shows the temporal aspects of a vectorized integration process. Here, only the external behavior (according to the start time T_0 and the end time T_1 of instances) must be serialized. The problem is defined as follows:

Definition 1. Integration Process Vectorization Problem (IPVP): *Let P denote a process plan and p_i with $p_i = (p_1, p_2, \ldots, p_n)$ denotes the process plan instances with*



(a) Instance-Based Process Plan P (b) Fully Vectorized Process Plan P'

Fig. 2. Vectorization of Integration Processes

$P \Rightarrow p_i$. Further, let each process plan P comprise a graph of operators $o_i = (o_1, o_2, \ldots, o_m)$. Due to serialization, the process plan instances are executed with $T_1(p_i) \leq T_0(p_{i+1})$. Then the integration process vectorization problem describes the search for the derived process plan P' that exhibits the highest degree of parallelism for the process plan instances p'_i such that the constraint conditions $(T_1(p'_i, o_i) \leq T_0(p'_i, o_{i+1})) \wedge (T_1(p'_i, o_i) \leq T_0(p'_{i+1}, o_i))$ hold and the semantic correctness is ensured.

Based on the IPVP, we investigate the static cost analysis, where in general, cost denotes the execution time. If we assume an operator sequence o with constant operator costs $C(o_i) = 1$, we get

$$\begin{aligned} C(P) &= n \cdot m && \text{// instance-based} \\ C(P') &= n + m - 1 && \text{// fully vectorized} \\ \Delta(C(P) - C(P')) &= (n-1) \cdot (m-1) \end{aligned}$$

where n denotes the number of process plan instances and m denotes the number of operators. Clearly, this is an idealized model, while typically lower improvements are reachable. Those depend on the most time-consuming operator o'_k with $C(o'_k) = \max_{i=1}^{m} C(o'_i)$ of a vectorized process plan P', where we get

$$\begin{aligned} C(P) &= n \cdot \sum_{i=1}^{m} C(o_i) \\ C(P') &= (n + m - 1) \cdot C(o'_k) \\ \Delta(C(P) - C(P')) &= n \sum_{i=1 \wedge i \neq k}^{m} C(o_i) - (n + m - 1) \cdot C(o'_k) \,. \end{aligned}$$

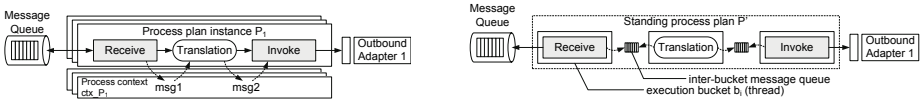
Obviously, $\Delta(C(P) - C(P'))$ can be negative in case of a very small n. However, with an increasing n, the performance improvement grows linearly.

2.3 Solution Overview

Here, we want to give a solution overview of our process plan vectorization approach. According to the generalized integration platform architecture, this exclusively addresses the process engine, while all other components can be reused without changes.

The core idea is to rewrite the instance-based process plan—where each instance is executed as a thread—to a vectorized process plan, where each operator is executed as a single *execution bucket* and hence, as a single thread. Thus, we model a standing process plan. Due to different execution times of the single operators, inter-bucket queues (with max constraints) are required for each data flow edge. Figure 3 illustrates those two execution models. Although significant performance improvement is possible, major challenges arise when rewriting P to P'.

Here, the main goal when rewriting a process plan P is the transparency of the used execution model. Hence, a user should only recognize the instance-based execution model, while internally (and transparent in the sense of being hidden to the user), the *vectorized execution model* is used. This aim poses several research challenges. This



(a) Instance-Based Process Plan P (b) Fully Vectorized Process Plan P'

Fig. 3. Different Execution Models

includes (1) ensuring *semantical correctness* of P', (2) preserving the *external behavior*, (3) ensuring *transactional behavior and recoverability*, and (4) realizing both the synchronous (simulated for P') as well as the asynchronous execution models. Finally, we must (5) handle the rewriting of different data flow concepts (from instance-based process plans, which use a variable-based data flow, to vectorized process plans that exhibit an explicit data flow (pipelining)). In order to overcome Problems 1-3, we present specific rewriting rules. Problem 4 is tackled with an extended message model. Finally, we propose operator-aware rewriting techniques in order to overcome Problem 5.

In the rest of the paper, we provide the details on how to rewrite an instance-based process plan into a vectorized process plan. Further, in Section 4, we present selected results of an exhaustive evaluation.

3 Rewriting Process Plans

In this section, we explain in detail how to rewrite instance-based process plans to fully vectorized process plans.

3.1 Message Model and Process Model

As formal foundation, we use the instance-based *Message Transformation Model (MTM)*. Hence, we have to define extensions in order to make it applicable also in the context of vectorized integration processes (then we refer to it as *VMTM*). Both consist of a message model and a process model.

We model a message m of a message type M as a quadruple with $m = (M, S, A, D)$, where M denotes the message type, S denotes the runtime state, and A denotes a map of atomic name-value attribute pairs with $a_i = (n, v)$. Further, D denotes a map of message parts, where a single message part is defined with $d_i = (n, t)$. Here, n denotes the part name and t denotes a tree of named data elements. In the VMTM, we extend it to a quintuple with $m = (M, C, S, A, D)$, where the context information C denotes an additional map of atomic name-value attribute pairs with $c_i = (n, v)$. This extension is necessary due to parallel message execution within one process plan.

A process plan P is defined with $P = (o, c, s)$ as a 3-tuple representation of a directed graph. Let o with $o = (o_1, \ldots, o_m)$ denote a sequence of operators, let c denote the context of P as a set of message variables msg_i, and let s denote a set of services $s = (s_1, \ldots, s_l)$. Then, an instance p_i of a process plan P, with $P \Rightarrow p_i$, executes the sequence of operators once. Each operator o_i has a specific type as well as an identifier NID (unique within the process plan) and is either of an *atomic* or of a *complex* type. Complex operators recursively contain sequences of operators with

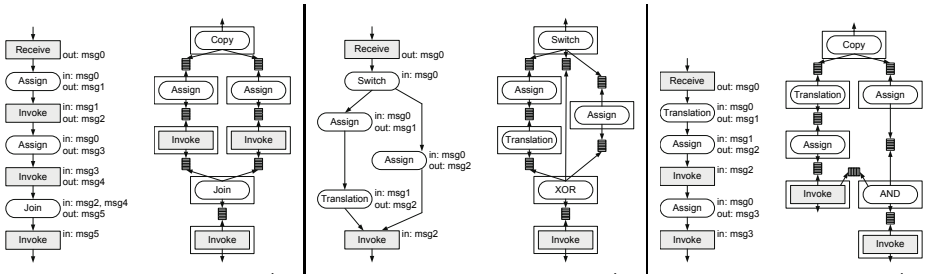
$o_i = (o_{i,1}, \ldots, o_{i,m})$. Further, an operator can have multiple input variables $msg_i \in c$, but only one output variable $msg_j \in c$. Each service s_i contains a type, a configuration and a set of operations. Further, we define a set of interaction-oriented operators *iop* (`Invoke`, `Receive` and `Reply`), control-flow-oriented operators *cop* (`Switch`, `Fork`, `Iteration`, `Delay` and `Signal`) and data-flow-oriented operators *dop* (`Assign`, `Translation`, `Selection`, `Projection`, `Join`, `Setoperation`, `Split`, `Orderby`, `Groupby`, `Window`, `Validate`, `Savepoint` and `Action`). Furthermore, in the *VMTM*, the flow relations between operators o_i do not specify the control flow but the explicit data flow in the form of message streams. Additionally, the `Fork` operator is removed due to redundancy. Finally, we introduce the additional operators `AND` and `XOR` (for synchronizing the serialized external behavior) as well as the `COPY` operator (for supporting the changed data flow).

3.2 Rewriting Algorithm

Now, let us focus on the realization of such process plan rewriting; even without considering transactional behavior and cost analysis, it is already very complex.

Algorithm 1. Process Plan Vectorization.

Require: operator sequence o
1: $B \leftarrow \emptyset, D \leftarrow \emptyset, Q \leftarrow \emptyset$
2: **for** $i = 1$ to $|o|$ **do**
3: // $\forall$ operators
4: **for** $j = i$ to $|o|$ **do**
5: // $\forall$ following operators
6: **if** $\exists o_i \xrightarrow{\delta} o_j$ **then**
7: $Q \leftarrow Q \cup q$ with $q \leftarrow$ create queue
8: $D \leftarrow D \cup d < o_i, q, o_j >$ with $d < o_i, q, o_j > \leftarrow$ create dependency
9: **end if**
10: **end for**
11: **if** $o_i \in Switch, Iteration, Fork, Savepoint, Invoke^*$ **then**
12: // see Subsubsections 3.2.2 and 3.2.3
13: **else**
14: $b_i(o_i) \leftarrow$ create bucket over o_i
15: **for** $k = 1$ to $|D|$ **do**
16: // foreach dependency
17: $d < o_x, q, o_y > \leftarrow d_k$
18: **if** $o_i \equiv o_x$ **then**
19: connect $b_i(o_i) \rightarrow q$
20: **else if** $o_i \equiv o_y$ **then**
21: connect $q \rightarrow b_i(o_i)$
22: **end if**
23: **end for**
24: $B \leftarrow B \cup b_i(o_i)$
25: **end if**
26: **end for**
27: **return** B



(a) Plan P (b) Plan P' (c) Plan P (d) Plan P' (e) Plan P (f) Plan P'

Fig. 4. Rewriting Examples (core concept, context-specific and serialized external behavior)

Rewriting Unary and Binary Operators. When rewriting instance-based process plans to vectorized process plans, we distinguish between unary operators (one input message: `Invoke`, `Assign`, `Translation`, `Selection`, `Projection`, `Split`, `Orderby`, `Groupby`, `Window`, `Action`, and `Delay`) and binary operators (multiple input messages: `Join`, `Setoperation`, and `Assign`). Both unary and binary operators can be rewritten with the same core concept (see Algorithm 1) that contains the following four steps. First, we create a queue instance for each data dependency between two operators (the output message of operator o_i is the input message of operator o_j with $j > i$). Second, we create an execution bucket for each operator. Third, we connect each operator with the referenced input queue. Clearly, each queue is referenced by exactly one operator, but each operator can reference multiple queues. Fourth, we connect each operator with the referenced output queues. If one operator must be connected to n output queues with $n \geq 2$ (its results are used by multiple following operators), we insert a `Copy` operator (gets a message from one input queue, then copies it $n-1$ times and puts those messages into the n output queues). In order to make the rewriting concept more understandable, we illustrate it using the following example.

Example 2. Vectorization of Unary and Binary Operators: *Assume a process plan P that receives a message, prepares two queries, loads data from two external sources, joins the results, and sends the final message to a third system (Figure 4(a)). If we vectorize this to P' (Figure 4(b)), we can apply the standard vectorization concept. The `Receive` operator has been removed because all operators directly read from queues. Further, the `Copy` operator has been inserted because both `Assign` operators have the same input. Additionally, there is the binary `Join` operator that reads messages from two concurrent input queues.*

Due to dependency checking, the process plan vectorization algorithm has a cubic worst-case complexity of $O(m^3) = O(m^3 + m^2)$.

Rewriting Context-Sensitive Operators. Now, we consider the context-specific operators `Switch`, `Iteration`, `Fork`, `Validate`, `Signal`, `Savepoint`, and `Reply`.

Rewriting `Switch` operators. When rewriting `Switch` operators, we must be aware of their ordered if-elseif-else semantics. Here, message sequences are routed along different switch-paths, which will eventually be merged. Assume a message sequence of

msg_1 and msg_2, where msg_1 is routed to path A, while msg_2 is routed to path B. If $C(A) \geq C(B) + C(Switch_B)$, msg_2 arrives earlier at the merging point than msg_1 does. Hence, a *message outrun* has taken place. Therefore, we have introduced the `XOR` operator that is inserted just before the single switch paths are merged. It reads from all queues (including a dummy queue for synchronization), compares the timestamps of read messages and forwards the oldest.

Example 3. Rewriting `Switch` Operators. *Assume a process plan P (Figure 4(c)). If we vectorize it to P' (Figure 4(d)), we apply the `Switch`-specific rewriting technique, where we create two pipeline branches (one for each switch-path). In order to avoid message outrun, we additionally inserted the `XOR` operator and a dummy queue.*

Rewriting `Iteration` operators. Also, when rewriting `Iteration` operators, the main problem is the *message outrun*. Here, we must ensure that all iteration loops (for a message) have been processed before the next message enters. Basically, a *for each* `Iteration` is rewritten to a sequence of (1) `Split` operator, (2) operators of the `Iteration` body and (3) `Setoperation` (union all) operator. In contrast to this, iterations with *while* semantics are not vectorized (one single execution bucket).

Rewriting `Validate` and `Signal` operators. One of the major differences between the instance-based process model and the vectorized process model is the maintenance of the process context (variables). Especially when dealing with validation, signals and error handling, this becomes crucial. Therefore, we extended the message model by context C (see Subsection 3.1). In case of an error (invalidity or explicit signal), we store the specific information in correlation to the current message that caused the signal. Then we can apply recovery processing.

In summary, when rewriting context-specific operators, we want to assure the semantic correctness during the rewriting of instance-based integration processes to the vectorized process model. This is a part of the general rewriting algorithm (Algorithm 1, lines 13-14). There are additional rewriting rules for `Fork`, `Savepoint` and `Reply` operators, which we omitted here because they are straight-forward.

Serialization and Recoverability. In order to realize the serialization of external behavior (precondition for transparency of the used execution model), we must ensure that explicitly modeled sequences of `Invoke` operators are serialized. Hence, we use the `AND` operator for synchronization purposes. If an `Invoke` operator has a temporal dependency, we insert an `AND` operator right before it as well as a dummy queue between the source of the temporal dependency and the `AND` operator. The `AND` operator reads from the dependency and the original queue and synchronizes the external behavior.

Example 4. Serialization of external behavior: *Assume a process plan P (Figure 4(e)). If we vectorize this process plan to P' (Figure 4(f)) with two pipeline branches, we need to ensure the serialized external behavior. Here, we insert an `AND` operator, where the left `Invoke` sends dummy messages to this operator. Only in the case that the right `Assign` as well as the left `Invoke` have been processed successfully, the real message of the right pipeline branch is forwarded to the second `Invoke`.*

With regard to recoverability of single integration processes, we might need to execute recovery processing with loaded queues. In general, we use the `stopped` flag of a

queue in order to stop it in case of a failure at operator o_i. In fact, we need to stop the input queue of this operator, while all other operators can continue working. Hence, the max queue constraint will be reached and clients are blocked.

Cost Analysis. In Subsection 2.2, we illustrated the theoretical performance of a simple sequence of operators, where each operator o_i has a single data dependency with the previous operator o_{i-1}. Now, we investigate the performance with regard to specific rewriting results (the idealized cost model is reused).

Parallel data flow branches. Here, different messages are processed by $|r|$ concurrent pipelines (branches) within P'. Examples for this are simply overlapping data dependencies and the `Switch` operator. Assume an operator sequence o of length m. In the instance-based model, the costs of n instances are $C(P) = n \cdot m$. In case the operator sequence contains a single branch with $|r| = 1$, we can improve the costs by $(n-1) \cdot (m-1)$ to $n+m-1$ using process plan vectorization. In the case of multiple branches with $|r| \geq 2$, the possible improvement is given by

$$C(P') = n + \max_{i=1}^{|r|}(|r_i|) - 1$$

$$\Delta(C(P) - C(P')) = n \cdot (m-1) - \max_{i=1}^{|r|}(|r_i|) + 1.$$

Clearly, in the case of $|r| = 1$ and $|r_1| = m$, the general cost analysis stays true. In the best case, $\max_{i=1}^{|r|}(|r_i|)$ is equal to $\frac{m}{|r|} \in \mathbb{N}$. The improvement is caused by the higher degree of parallelism. However, parallel data-flow branches may also cause overhead for splitting (`Copy`) and merging (`AND` or `XOR`).

Rolled-out `Iteration`. When rewriting `Iteration` operators with *for each* semantics, we split messages according to the *for each* condition and process the iteration body as inner pipeline without cyclic dependencies. Finally, the processed sub-messages are merged using the `Setoperation` operator (union all). In the instance-based case, $C(o) = r \cdot m$ is true, where r denotes the number of iteration loops (number of sub-messages) and m denotes the number of operators in the iteration body. Due to the sub-pipelining, we can reduce the processing time to $C(o') = r + m - 1 + 2$.

3.3 Cost-Based Vectorization

The two major weaknesses of our approach are (1) that the theoretical performance of a vectorized integration process mainly depends on the performance of the most cost-intensive operator, and (2) that the practical performance also strongly depends on the number of available threads. Thus, the optimality of vectorization strongly depends on dynamic workload characteristics. Hence, future work should investigate the generalized problem description, where we search for the optimal k execution buckets (each containing a number of operators) in a cost-based manner.

4 Experimental Evaluation

In this section, we provide selected experimental results. Basically, we can state that the vectorization of integration processes leads to a significant performance improvement for different scale factors.

4.1 Experimental Setup

We implemented the introduced approaches within the so-called WFPE (workflow process engine) using Java 1.6 as the programming language. This implementation is available upon request. In general, the WFPE uses compiled process plans (a java class is generated for each integration process type). Furthermore, it follows an instance-based execution model. Now, we integrated components for the static vectorization of integration processes (we call this VWFPE). For that, new deployment functionalities were introduced (those processes are executed in an interpreted fashion) as well as several changes in the runtime environment were realized.

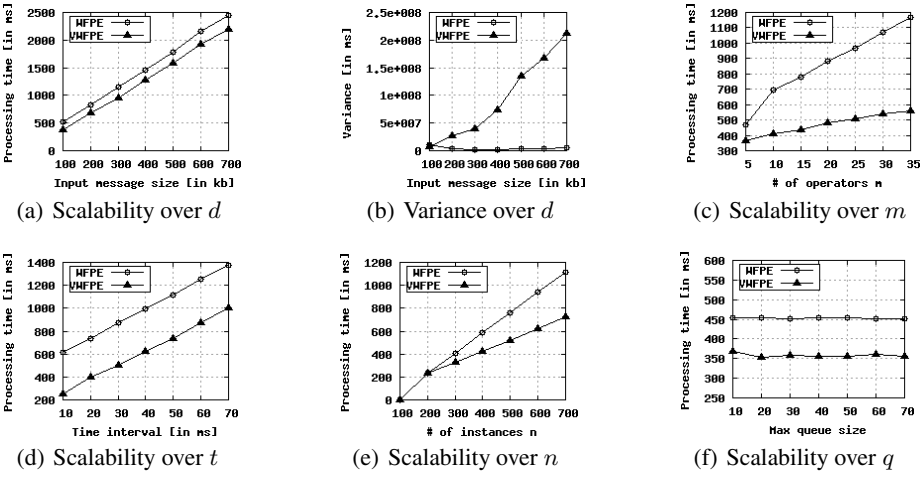
We ran our experiments on a standard blade (OS Suse Linux) with two processors (each of them a Dual Core AMD Opteron Processor 270 at 1,994 MHz) and 8.9 GB RAM. Further, we executed all experiments on synthetically generated XML data (using the DIPBench toolsuite [2]). In general, we used the following five aspects as scale factors: data size d of a message, the number of operators m of a process plan, the time interval t between two messages, the number of process instances n and the maximal number of messages q in a queue. Here, we measured the performance of different combinations of those. For statistical correctness, we repeated all experiments 20 times.

As base integration process for our experiments, we used a sequence of six operators. Here, a message is received (Receive) and then an interaction is prepared (Assign) and executed with the file adapter (Invoke). After that, the resulting message (contains orders and orderlines) is translated using an XML transformation (Translation) and finally sent to a specific directory (Assign, Invoke). We refer to this as $m = 5$ because the Receive is removed during vectorization. When scaling m up to $m = 35$, we copy and reconfigure those operators.

4.2 Performance and Throughput

Here we ran a series of experiments based on the already introduced scale factors. The results of these experiments are shown in Figure 5.

In Figure 5(a) we scaled the data size d of the input messages from 100kb to 700kb XML messages and measured the processing time for 250 process instances ($n = 250$) needed by the three different runtimes. There, we fixed $m = 5$, $t = 0$, $n = 250$ and $q = 50$. We can observe that both runtimes exhibit a linear scaling according to the data size and that significant improvements can be reached using vectorization. There, the absolute improvement increases with increasing data size. Further, in Figure 5(b), we illustrated the variance of this sub-experiment. The variance of the instance-based execution is minimal, while the variance of the vectorized runtime is worse because of the operator scheduling. Now, we fixed $d = 100$ (lowest absolute improvement in 5(a)), $t = 0$, $n = 250$ and $q = 50$ in order to investigate the influence of m. We varied m from 5 to 35 operators. Interestingly, not only the absolute but also the relative improvement of vectorization increases with increasing number of operators. Figure 5(d) shows the impact of the time interval t between the initiation of two process instances. For that, we fixed $d = 100$, $m = 5$, $n = 250$, $q = 50$ and varied t from 10ms to 70ms. The absolute improvement between instance-based and vectorized approaches decreases slightly with increasing t. As an explanation, the time-interval has no impact on the instance-based execution. In contrast to that, the vectorized approach depends on



(a) Scalability over d (b) Variance over d (c) Scalability over m

(d) Scalability over t (e) Scalability over n (f) Scalability over q

Fig. 5. Evaluation Results for Experimental Performance

t due to the resource scheduling whenever not all of the execution buckets need CPU time. Further, we analyze the influence of the number of instances n as illustrated in Figure 5(e). Here, we fixed $d = 100$, $m = 5$, $t = 0$, $q = 50$ and varied n from 100 to 700. Basically, we can observe that the relative improvement between instance-based and vectorized execution increases with increasing n, due to parallelism of process instances. Figure 5(f) illustrates the influence of the maximal queue size q, which we varied from 10 to 70. Here, we fixed $d = 100$, $m = 5$, $t = 0$ and $n = 250$. In fact, q slightly affects the overall performance for a small number of concurrent instances n. However, at $n = 250$, we cannot observe any significant influence according to the performance for both approaches.

5 Related Work

Database Management Systems. In the context of DBMS, throughput optimization has been addressed with different techniques. One significant approach is data sharing across common subexpressions of query instances [3,4]. However, in [5] it was shown that sharing can also hurt performance. Another inspiring approach is given by staged DBMS [6]. Here, in the QPipe Project [7], each relational operator was executed as a micro-engine (one operator, many queries). Additional approaches exist in the context of distributed query processing [8,9].

Data Stream Management Systems. Further, in data stream management systems (DSMS) and ETL tools, the *pipes and filters* execution model is widely used. Examples for those systems are QStream [10], Demaq [11] and Borealis [12]. However, in DSMS, scheduling is not realized with multiple processes or threads but with central control strategies and thus, the problems addressed in this paper are not present.

Streaming Service and Process Execution. In service-oriented environments, throughput optimization has been addressed on different levels. Performance and resource

issues, when processing large volumes of XML documents, lead to message chunking on the service-invocation level. There, request documents are divided into chunks, and services are called for every single chunk [13]. An automatic chunk-size computation using the extremum-control approach was addressed in [14]. On the process level, pipeline scheduling was incorporated in [15] into a general workflow model to show the valuable benefit of pipelining in business processes. Further, [16] add pipeline semantics to classic step-by-step workflows.

Integration Process Optimization. This has not yet been explored sufficiently. There are platform-specific optimization approaches for the *pipes and filters* execution model, like the optimization of ETL processes [17]; there are also numerous optimization approaches for instance-based processes like the optimization of data-intensive decision flows [18], the static optimization of the control flow using critical path approaches [19] and SQL-supporting BPEL activities and their optimization [20]. Further, the execution time minimization of integration processes [21] was already investigated.

6 Conclusions

In order to optimize the throughput of integration platforms, in this paper, we introduced the concept of automatic vectorization of integration processes. We showed how integration processes can be rewritten in a transparent manner, where the internal execution model is hidden from the user in order to reach a higher degree of parallelism while ensuring the transactional behavior and external behavior similar to instance-based integration processes. Based on our experimental evaluation, we can state that significant throughput improvement is possible and the concept of process vectorization is applicable in practice. Future work should address the cost-based vectorization.

References

1. Boehm, M., Habich, D., Lehner, W., Wloka, U.: An advanced transaction model for recovery processing of integration processes. In: ADBIS (2008)
2. Boehm, M., Habich, D., Lehner, W., Wloka, U.: Dipbench toolsuite: A framework for benchmarking integration systems. In: ICDE (2008)
3. Dalvi, N.N., Sanghai, S.K., Roy, P., Sudarshan, S.: Pipelining in multi-query optimization. In: PODS (2001)
4. Roy, P., Seshadri, S., Sudarshan, S., Bhobe, S.: Efficient and extensible algorithms for multi query optimization. In: SIGMOD (2000)
5. Johnson, R., Hardavellas, N., Pandis, I., Mancheril, N., Harizopoulos, S., Sabirli, K., Ailamaki, A., Falsafi, B.: To share or not to share? In: VLDB (2007)
6. Harizopoulos, S., Ailamaki, A.: A case for staged database systems. In: CIDR (2003)
7. Harizopoulos, S., Shkapenyuk, V., Ailamaki, A.: Qpipe: A simultaneously pipelined relational query engine. In: SIGMOD (2005)
8. Ives, Z.G., Florescu, D., Friedman, M., Levy, A.Y., Weld, D.S.: An adaptive query execution system for data integration. In: SIGMOD (1999)
9. Lee, R., Zhou, M., Liao, H.: Request window: an approach to improve throughput of rdbms-based data integration system by utilizing data sharing across concurrent distributed queries. In: VLDB (2007)

10. Schmidt, S., Berthold, H., Lehner, W.: Qstream: Deterministic querying of data streams. In: VLDB (2004)
11. Boehm, A., Marth, E., Kanne, C.C.: The demaq system: declarative development of distributed applications. In: SIGMOD (2008)
12. Abadi, D.J., Ahmad, Y., Balazinska, M., Çetintemel, U., Cherniack, M., Hwang, J.H., Lindner, W., Maskey, A., Rasin, A., Ryvkina, E., Tatbul, N., Xing, Y., Zdonik, S.B.: The design of the borealis stream processing engine. In: CIDR (2005)
13. Srivastava, U., Munagala, K., Widom, J., Motwani, R.: Query optimization over web services. In: VLDB (2006)
14. Gounaris, A., Yfoulis, C., Sakellariou, R., Dikaiakos, M.D.: Robust runtime optimization of data transfer in queries over web services. In: ICDE (2008)
15. Lemos, M., Casanova, M.A., Furtado, A.L.: Process pipeline scheduling. J. Syst. Softw. 81(3) (2008)
16. Biornstad, B., Pautasso, C., Alonso, G.: Control the flow: How to safely compose streaming services into business processes. In: IEEE SCC (2006)
17. Simitsis, A., Vassiliadis, P., Sellis, T.: Optimizing etl processes in data warehouses. In: ICDE (2005)
18. Hull, R., Llirbat, F., Kumar, B., Zhou, G., Dong, G., Su, J.: Optimization techniques for data-intensive decision flows. In: ICDE (2000)
19. Li, H., Zhan, D.: Workflow timed critical path optimization. Nature and Science 3(2) (2005)
20. Vrhovnik, M., Schwarz, H., Suhre, O., Mitschang, B., Markl, V., Maier, A., Kraft, T.: An approach to optimize data processing in business processes. In: VLDB (2007)
21. Boehm, M., Habich, D., Lehner, W., Wloka, U.: Workload-based optimization of integration processes. In: CIKM (2008)

Invisible Deployment of Integration Processes

Matthias Boehm[1], Dirk Habich[2], Wolfgang Lehner[2], and Uwe Wloka[1]

[1] Dresden University of Applied Sciences, Database Group
mboehm@informatik.htw-dresden.de,
wloka@informatik.htw-dresden.de
[2] Dresden University of Technology, Database Technology Group
dirk.habich@tu-dresden.de,
wolfgang.lehner@tu-dresden.de

Abstract. Due to the changing scope of data management towards the management of heterogeneous and distributed systems and applications, integration processes gain in importance. This is particularly true for those processes used as abstractions of workflow-based integration tasks; these are widely applied in practice. In such scenarios, a typical IT infrastructure comprises multiple integration systems with overlapping functionalities. The major problems in this area are high development effort, low portability and inefficiency. Therefore, in this paper, we introduce the vision of *invisible deployment* that addresses the virtualization of multiple, heterogeneous, physical integration systems into a single logical integration system. This vision comprises several challenging issues in the fields of deployment aspects as well as runtime aspects. Here, we describe those challenges, discuss possible solutions and present a detailed system architecture for that approach. As a result, the development effort can be reduced and the portability as well as the performance can be improved significantly.

Keywords: Invisible deployment, Integration processes, Virtualization, Deployment, Optimality decision, Heterogeneous integration platforms.

1 Introduction

Integration processes—as an abstraction for workflow-based integration tasks—gain in importance because data management continuously changes towards the management of distributed and heterogeneous systems and applications. There, the performance of complete IT infrastructures depends on the central integration platforms. In this context, different integration system types are used. Examples for those types are Federated DBMS, EAI (Enterprise Application Integration) servers, ETL (Extraction Transformation Loading) tools, and WfMS (Workflow Management Systems). However, the boundaries between these different classes of systems begin to blur due to overlapping functionalities of concrete products. Major problems in this context are posed by the high development effort for integration task specification, the low degree of portability between those integration systems, and the possible inefficiency. The inefficiency problem (optimization potential) is caused by system-inherent assumptions about the primary application context. If the actual workload characteristics (process types, data

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 53–65, 2009.
© Springer-Verlag Berlin Heidelberg 2009

size) differ from those assumptions, the execution performance can be significantly improved by changing the used integration system.

Our main hypotheses are (1) that a typical IT infrastructure comprises multiple integration systems with overlapping functionalities, and (2) that we can generate platform-specific integration task specifications from platform-independent models. The opportunities arising from these hypotheses led us to our idea of *invisible deployment*. Here, a user models an integration process in a platform-independent way and deploys it using a central deployment interface. Now, there is the general possibility to decide on the optimal integration platform to execute the specified integration process. This decision should consider workload execution statistics in order to be based on objective online statistics with regard to changing workload characteristics. Clearly, this general idea can overcome the problems of high development effort and low portability (generation of process descriptions) as well as inefficiency (optimality decision, load balancing), but it comes with several inherent challenges.

In order to overcome the described problems and to convey the core idea of *invisible deployment*, in this paper, we make the following contributions.

- In Section 2, we introduce the vision of *invisible deployment* and describe the major challenges that arise when realizing that vision.
- Subsequently, in Section 3, we describe our approach for the deployment of integration processes. Here, we focus on the selected aspects of integration process generation and functional candidate set determination.
- Furthermore, in Section 4, we discuss a possible runtime approach, where we investigate cost modeling and cost normalization, optimality decisions and the heterogeneous load balancing.
- Based on the proposed solution, in Section 5, we present a system architecture for the realization of *invisible deployment* in the context of integration platforms.

Finally, we survey related work in Section 6 and conclude the paper in Section 7.

2 Vision Overview

Based on the described problems, in this section, we pose our main hypotheses and present the resulting conceptual architecture for the vision of *invisible deployment*. Additionally, we point out the major challenges that arise here.

2.1 Assumptions and Hypotheses

In fact, our vision is based on empirically evaluated assumptions. Here, we conclude the following two hypotheses.

Hypothesis 1. Model-Driven Generation: *Integration processes can be modeled in a platform-independent way. Based on those models, we can generate proprietary (platform-specific) integration task specifications for concrete integration platforms.*

Hypothesis 2. Optimality Decision: *A typical IT infrastructure comprises multiple integration systems with overlapping functionalities. Hence, there is the possibility to decide on the optimal integration system based on functional and non-functional properties according to given integration processes.*

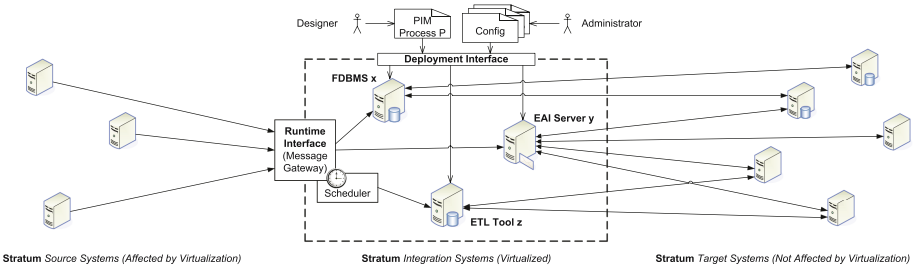


Fig. 1. Vision of *Invisible Deployment*

Here, the model-driven generation of integration processes is the precondition for the *invisible deployment*. It has been shown in several projects (GCIP project [1,2], Orchid project [3], and ETL process management [4]) that process generation can be done using concepts from the model-driven software development. Furthermore, a typical IT infrastructure comprises multiple integration systems with overlapping functionalities [5] such as specific operators, supported external systems, possibilities to react to external events or transactional functionalities. Hence, an integration processes can be executed using different integration systems, without changing the external behavior. Thus, the decision on the optimal integration system can be made based on non-functional properties such as efficiency, scalability and resource consumption.

2.2 Conceptual Architecture

If there are multiple integration systems with overlapping functionalities, we can decide on the optimal system for given integration processes. Usually, this decision is made based on subjective experience and certain workload assumptions. Hence, there is the need for a conceptual architecture that allows for the objective optimality decision as well as for transparency (hiding) of the used integration system. Clearly, this is a virtualization approach. In contrast to typical virtualization, where multiple logical systems are mapped to one physical system, the *invisible deployment* addresses the inverse problem, where one logical system has to be mapped to multiple physical systems. In fact, this is similar to shared-nothing architectures, where only meta data (about the distribution) has to be centrally maintained.

Figure 1 shows the conceptual architecture for our vision of *invisible deployment*. Here, we have to distinguish three strata: the stratum of source systems, the stratum of integration systems and the stratum of target systems. Clearly, a source system can be a target system at the same time. Furthermore, we have two major types of integration processes that are deployed into and executed by the integration systems. First, there are data-driven integration processes where instances are initiated by messages sent from the source systems to the integration systems (synchronous as well as asynchronous execution model). Second, there are scheduled integration processes that are initiated by an internal time-based scheduler of the integration system.

The vision of *invisible deployment* describes the transparent deployment of integration processes into a logical integration system that consists of a set of heterogeneous

physical integration systems. In order to realize this kind of transparency, we need to focus on deploytime and runtime challenges. Assume that the designer has modeled integration processes in a platform-independent way (PIM); then we need to determine the subset of integration systems that can realize the modeled integration process. Further, we need to generate platform-specific integration tasks based on the platform-independent model. Those aspects are realized by the deployment interface. In order to allow for transparency, we now focus on the runtime requirements. Here, we must provide a runtime interface for the source systems. Thus, we do not specify the physical integration systems as target systems for message propagations but we specify the single logical integration system. Furthermore, the transparency for external target systems is pretty simple because here, we only need to distribute the configurations to all used physical integration systems. This runtime interface and a central scheduler for all physical integration systems allow for the optimality decision on used integration systems. Therefore, we need to monitor execution statistics, normalize those costs into a platform-independent cost model and allow for cost estimation over heterogeneous physical integration systems. In fact, this opens the opportunity for optimization approaches such as dynamic optimality decisions and heterogeneous load balancing.

2.3 Problems and Challenges

After having introduced the conceptual architecture, we now want to focus on the major challenges that arise with regard to process deployment and runtime scheduling.

Deployment Challenges. The deployment challenges address the generation of platform-specific integration tasks and the deployment into the integration systems.

Challenge 1. Generation of Integration Processes: *The precondition for the* invisible deployment *is the platform-independent modeling of integration processes and the generation of platform-specific integration tasks. Especially, the bi-directional model transformation without information loss poses a fundamental challenge.*

Challenge 2. Functional Candidate Set Determination: *Based on Challenge 1, there is the need to evaluate whether or not a given integration process can be executed with a specific integration system. Here, we need to derive functional requirements from the integration process and match those with feature sets of the specific integration systems.*

Challenge 3. Configuration Management: *Due to the virtualization of multiple heterogeneous physical integration systems (each with its own configuration), also the platform-independent configuration management is a tough challenge. Here, the specification of transactional behavior and the adapter/wrapper configurations are important. In particular, this is required for the application in practice.*

Challenge 4. Reliability: *On top of candidate set determination and configuration management, there is the need to ensure functional (semantical) correctness. Hence, model checking techniques must be investigated in order to prove the conformance of certain integration system configurations with the semantics of specified integration processes.*

Runtime Challenges. In addition to the deployment challenges, we further see the following runtime challenges. In particular, the platform-independent cost modeling as well as the serialization and transactional behavior are important.

Challenge 5. Cost Modeling and Cost Normalization: *In order to adapt to changing workload characteristics and allow for comparison over heterogeneous integration systems, we need to monitor execution statistics. There is a need for a platform-independent cost model and cost normalization approaches to allow for comparability.*

Challenge 6. Optimality Decision: *Based on the comparability of integration systems, we will be able to decide on the optimal integration system for a given integration process. Here, a periodical re-decision seems to be advantageous. In fact, such a decision can be made for one given process, for a set of k given processes or for arbitrary subgraphs of processes. Clearly, this is a challenging combinatoric problem.*

Challenge 7. Heterogeneous Load Balancing: *The heterogeneous load balancing extends the challenge of the optimality decision with the aim of an optimal utilization of all physical integration systems rather than only finding the optimal integration system. Here, different optimization objectives are present.*

Challenge 8. Serialization and Transactional Behavior: *Although the heterogeneous load balancing has high optimization potential, it poses a problem of serialization and transactional behavior. If a sequence of two messages is forwarded to different physical integration systems, we need to serialize those process executions in order to prevent the message outrun (avoid changing external behavior).*

In order to be concise, we try to highlight core ideas of possible solutions for a selected subset of those challenges in the following two sections.

3 Deployment

In this section, we want to explain possible solutions for selected deployment challenges. In fact, the generation of integration processes and the functional candidate set determination are the most important deployment aspects.

3.1 Integration Process Generation

The generation of integration processes has been investigated intensively. The GCIP project (Generation of Complex Integration Processes) [1,2] and the correlated GCIP Framework allow for the platform-independent modeling of integration processes as well as the generation and optimization of platform-specific integration tasks for concrete integration systems. Figure 2 illustrates the general GCIP approach as well as its current project state. In general, the generation framework comprises the four layers of certain platform-independent models (PIM), a central abstract platform-specific model (A-PSM), platform-specific models (PSM) and the declarative process descriptions (DPD). Currently, the framework supports five concrete integration systems (for the types FDBMS, ETL and EAI) as target of our generation.

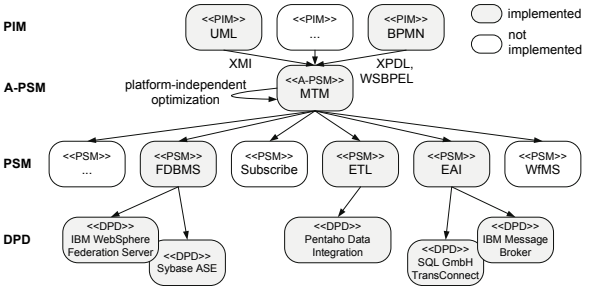


Fig. 2. Overview of GCIP Generation Approach

On a platform-independent level, integration processes can be modeled with different languages, such as UML (Unified Modeling Language) activity diagrams or BPMN (Business Process Modeling Notation) process specifications. In fact, those specifications are directed graphs that can be annotated in order to provide details and other parameters. Finally, those models can be imported and transformed to an abstract platform-specific model. In the case of UML, XMI documents are imported, while in the case of BPMN, we can use XPDL as well as WSBPEL specifications, respectively.

In contrast to typical model-driven approaches, a central abstract platform-specific model has been introduced, where the *Message Transformation Model* [6] and its defined operators are used. This central model is independent from any integration system type and reduces the transformation complexity between m PIMs and n PSMs from $m \cdot n$ to $m + n$; additionally, it provides the possibility to apply platform-independent optimization techniques.

Based on the A-PSM, platform-specific models can be generated. Those models are specific to the integration system type but not specific to the concrete integration systems. Currently, the groups of FDBMS, EAI and ETL are supported. Here, XML representations are used for those internal models.

Finally, declarative process descriptions are generated from the single platform-specific models. In the model-driven architecture, this is called the `code` layer. However, we explicitly separate this from the internal code layer of the integration systems. Hence, we use the name DPD. As an example, we can generate stored procedures (scheduled integration processes) or triggers (data propagations) of different SQL dialects. Further, also EAI processes (message flows) and ETL jobs can be generated.

In conclusion, integration tasks for concrete integration systems can be generated based on platform-independent specifications. For *invisible deployment*, this is the foundation for all other subsequent challenges. In fact, it has been shown that this is realizable for fully different types of integration systems.

3.2 Candidate Set Determination

Based on the generation of integration processes, we need to determine a candidate set in the sense of a subset of integration systems that are able to realize a given integration process P (Challenge 2). Hence, we determine candidates for the optimal integration system. Therefore, a two-phase approach is meaningful. In the first phase, the functional

requirements are derived from the given integration process (e.g., the ability to receive messages). Based on this feature set $F(P)$ and the defined feature sets of all supported integration systems $F(S)$, in the second phase, a matching between the single feature sets is computed in order to determine the logical candidate set C of integration systems that can be used to execute P.

Algorithm 1 illustrates the details of the system candidate set determination. First, for each operator o_i with $o_i \in P$, the functional requirements are derived and added to the feature set $F(P)$ (lines 3-5). Then, deployment policies $D(P)$—such as the choice of execution model (synchronous/asynchronous) or certain transactional requirements—are also added to the feature set (lines 6-8). After this first algorithm phase, the second phase realizes the matching of feature sets. Therefore, we iterate over all systems s_i and the single feature sets $F(s_i)$ in order to compare those features with the feature set $F(P)$. If we determine two equal features, we set $fflag$ to true (line 16). Further, if we have determined that there are no equal features, $sflag$ is set to false (line 21) and the current system s_i is not included into the candidate set C. Finally, C contains all

Algorithm 1. Candidate Set Determination.

Require: process plan P, deployment policy $D(P)$, feature sets $F(S)$, system set S

```
 1: C ← ∅, F(P) ← ∅
 2: // part 1: derive functional properties of P
 3: for i = 1 to |P| do                                  // foreach operator o_i
 4:    F(P) ← F(P) ∪ f(o_i)
 5: end for
 6: for i = 1 to |D(P)| do                               // foreach policy dp_i
 7:    F(P) ← F(P) ∪ f(dp_i)
 8: end for
 9: // part 2: match feature sets
10: for i = 1 to |S| do                                  // foreach system s_i
11:    for j = 1 to |F(s_i)| do                          // foreach system feature f_j
12:       sflag ← true
13:       fflag ← false
14:       for k = 1 to |F(P)| do                        // foreach plan feature f_k
15:          if f_j = f_k then
16:             fflag = true
17:             break 14
18:          end if
19:       end for
20:       if NOT fflag then
21:          sflag = false
22:          break 11
23:       end if
24:    end for
25:    if sflag then
26:       C ← C ∪ s_i
27:    end if
28: end for
29: return C
```

systems that fully conform to the required functionalities. Clearly, this algorithm has a worst-case (in the case of $C = S$) complexity of $O(\sum_{i=1}^{|S|} (|F(s_i)| \cdot |F(P)|))$, where $|F(P)| = m + |D(P)|$ and m denotes the number of operators.

4 Runtime

Aside from the deployment aspects, there are also runtime challenges, and we want to use this section to explain possible solutions for them. Here, we discuss the cost modeling as well as static and dynamic optimality decisions.

4.1 Platform-Independent Cost Model and Cost Normalization

In fact, if we have multiple physical candidate integration systems and if we want to decide on the optimal integration system, there is a need for a platform-independent cost model as well as algorithms for the cost normalization of monitored statistics into that model. We can only normalize statistics from platform-specific models into the platform-independent model but not vice versa because there is an infinite number of denormalized forms of one normalized form. Actually, our cost model contains two types of costs: the abstract cost $C(P)$ that is defined by cardinality formulas as well as the weighted cost $C'(P)$ that is computed by $C'(P) = \frac{t_e(P)}{C(P)}$ as the ratio of execution time $t_e(P)$ and abstract cost $C(P)$. Hence, we can overcome the problem of possible different hardware and disjoint process instances in the sense of computing tuple rates. In fact, we only need to monitor and normalize cardinality and execution time statistics at the operator granularity.

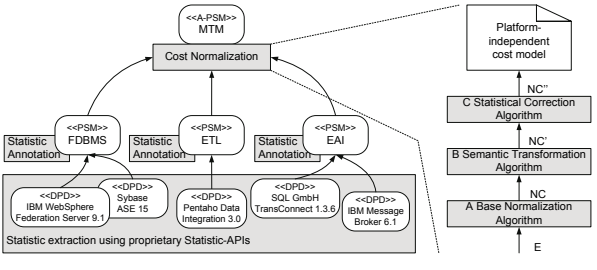


Fig. 3. Cost Normalization Overview

Figure 3 illustrates the general concept of statistic extraction and its normalization into the described platform-independent model. Execution statistics (cardinalities and execution times) are extracted using the system-specific statistic APIs of the physical integration systems. Those statistics are annotated at PSM level and finally mapped to the platform-independent cost model. This mapping comprises the cost normalization. Here, we use three algorithms in order to overcome the sub-challenges of cost normalization. The *base normalization algorithm* overcomes the problems of parallelism of process instances, different resource utilization and different execution models by

computing normalized part times (execution time, number of parallel instances, effective and maximal resource allocation). Further, the *semantic transformation algorithm* overcomes the problem of different semantics of extracted statistics. Here, we need to be aware of 1:1, 1:N, N:1 and N:M mappings between platform-specific and platform-independent operators. Obviously, for 1:N mappings and N:M mappings, we cannot aggregate the measured statistics and hence, the result is missing statistics for parts of the process. Finally, there is the *statistical correction algorithm*. It overcomes the problems of inconsistent statistics and missing statistics by checking operator sequences as well as computing missing statistics by linear extrapolation.

Finally, we can use the platform-independent cost model as well as the normalized statistics in order to decide on the optimal integration system in a cost-based fashion (aware of workload characteristics).

4.2 Optimality Decision

The optimality decision addresses the static decision on the optimal integration system, similar to an advisor decision. In conclusion of such a decision, a given integration process should be executed with this integration system.

In order to do so, we need to deploy the integration process P into all candidate integration systems s_i with $s_i \in C$. Then, we require several reference runs of P on each of those systems in order to gather statistics. Subsequently, we can normalize the statistics as described and finally, we decide on the optimal integration system. In order to adapt to changing workload characteristics, we need to execute those reference runs periodically. In fact, the application areas for such a decision are mainly scheduled integration processes or static decisions for data propagations.

The most obvious problem in that context is the optimality decision on exactly one integration process P. If we generalize this problem, we need to decide on a set of integration processes $k \cdot P$. Clearly, here we can choose one integration system for all k integration processes (trade-off between different processes) or use the simple decision problem for each of those processes. Furthermore, we can also consider different combinations of subgraphs of all integration processes. Clearly, we get an exponential number of alternative distributions to decide on.

Obviously, the major problem of this static optimality decision is that we do not utilize all resources (hardware in the case of physically separated systems). In fact, we always use the optimal integration system (which may change over time), but we do not use multiple systems at the same time.

4.3 Heterogeneous Load Balancing

In order to overcome the previously mentioned problem of suboptimal resource utilization, we introduce the concept of heterogeneous load balancing over multiple heterogeneous physical integration systems. Hence, this results in a dynamic and continuous optimality decision. Here, we preferably use the optimal integration system but the other candidate systems s_i with $s_i \in C$ may be used as well. Therefore, the optimality decision is changed from a deployment approach (periodically re-executed) to a dynamic runtime approach where the optimality decision must be made continuously.

In this context, the major problem is the assurance of serialization and transactional behavior with respect to the serialization according to the incoming message order and the observable external behavior. Assume, for instance, a process type P (with process instances p_i) that executes messages in sequential order; thus, we have to ensure the serialized order of $end(p_i) \leq start(P_{i+1})$. If we distribute those messages to two different integration systems that use the asynchronous execution model, the anomaly problem of message outrun [7] can occur. If the integration systems use the synchronous execution model, the serialization of process instances is simple but inefficient because—due to scheduling overhead—the resource utilization of the overall architecture is even worse than for the static optimality decision.

Our general solution for this heterogeneous load balancing problem is the distribution of fully disjoint message sequences according to their correlated integration processes across multiple physical integration systems. Therefore, we need to predict the costs in a platform-independent manner once more but now, we also need to predict the future workload and we must solve the balancing problem. This problem comprises the search for the optimal distribution of k integration process types across $|C|$ integration systems s_i such that the globally optimal solution is used (with regard to the optimization objectives (1) throughput maximization, (2) latency minimization or (3) load balance maximization). Clearly, we can extend this to a more fine-grained decision model, where subgraphs (similar to the challenge of optimality decision) are distributed across the integration systems.

If the workload changes over time, we need to exchange the execution context between integration systems. Hence, our optimality decision must be aware of the costs that are necessary for switching integration systems (due to synchronization efforts).

5 System Architecture

Figure 4 illustrates an architecture realizing the vision of *invisible deployment*. Basically, the message propagation is supported by an execution interface. Further, the integration task specifications $PD_i(x)$ (process types, configurations) are also possible using a deployment interface. For deployment purposes, process transformers as well as process deployers are required to generate platform-specific models and to deploy those into the different integration systems used. Furthermore, an `Optimizer` component for rewriting is needed. Deployed processes are registered within a central `Repository`. Here, all types of systems, except for client systems, as well as the time schedules are managed. Concerning the execution of the integration tasks, the synchronous events are directly forwarded to the `Runtime Environment`, while asynchronous events are appended to a specific `Request Queue`. Independent from this, the `Scheduler` also invokes the `Runtime Environment` directly, based only on the defined time schedules. Within the `Core Execution Service`, the integration task is split into subtasks. Here, the transactional behavior is ensured as well. The `Dispatcher` decides about the optimal integration system for each integration task and invokes the registered integration systems via `IS Gateways`. These decisions are based on functional as well as non-functional properties, including monitored workload characteristics. Finally, such an environment provides a central deployment

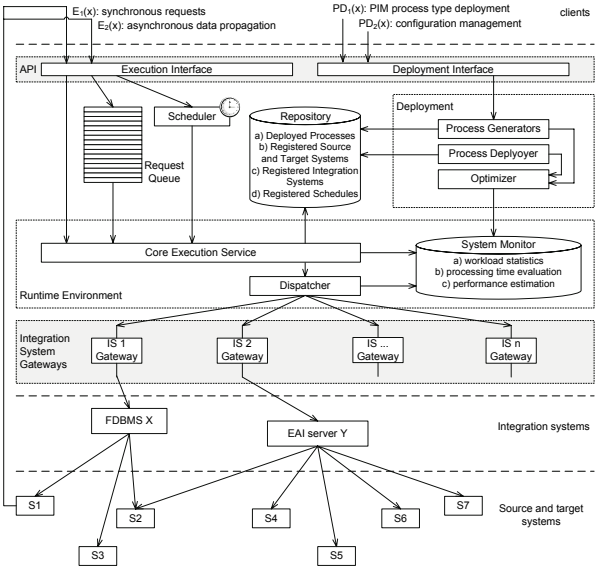


Fig. 4. Hybrid Gateway EIP Micro-Architecture

of integration tasks, using a distributed system infrastructure (of integration systems) during execution. This is the core requirement of a service-oriented architecture (SOA). Finally, this type of gateway integration system can realize the *invisible deployment* in a transparent manner.

6 Related Work

In fact, the *invisible deployment* is a novel vision and no comparable work exists. Here, we want to survey application areas as well as correlated virtualization approaches.

6.1 Application Areas

In the context of the generation and deployment of integration processes, we need to emphasize three projects and approaches, respectively.

The Orchid project [3] addresses the generation of ETL jobs based on declarative mapping specifications. There, a so-called Operator Hub Model (OHM) is used in order to transform the execution semantics of ETL jobs into a platform-independent form. The Orchid project is restricted to the generation of ETL processes for IBM ETL tools. In general, an extension to vendor-independent semantics seems to be possible without conceptual problems. While Orchid addresses only the generation of ETL jobs, the approach presented in [4] focuses on the combination of ETL process generation and model management. There, the authors presented platform-independent operators for the deployment of ETL processes. In contrast to those two approaches, the GCIP (Generation of Complex Integration Processes) Framework focuses on the modeling of platform-independent integration processes [1], the generation for numerous different

integration system types (such as FDBMS, EAI servers and ETL tools) as well as the application of optimization techniques [2,8] during model-driven generation.

The major similarity of those three projects is the possibility of deciding on the optimal target integration system to use (target of the generation). Hence, the vision of *invisible deployment* can be applied to all of those approaches.

6.2 Virtualization Approaches

Clearly, the *invisible deployment* is a virtualization approach. In the context of *software as a service*, in particular, the database virtualization seems advantageous. Here, several approaches exist for the realization of multi-tenant databases [9,10] where multiple logical databases are maintained within one physical database. Obviously, there are more general approaches for multi-tenant software [11] as well as for IT service provision [12]. By now, the famous term *cloud computing* [13] has been established for a superset of those virtualization approaches. Even more, there is an approach [14] on how to provide EAI as a service. Those approaches virtualize multiple logical systems into one single physical system. In contrast to this, according to the scalability terminology [15], we virtualize one logical integration system into a farm of heterogeneous, physical integration systems.

7 Summary

To summarize, in this paper, we introduced our novel vision of *invisible deployment* that is applicable in many different areas and that exhibits a high optimization potential as well as numerous challenging research aspects. In general, the *invisible deployment* is based on the hypothesis that a typical IT infrastructure comprises multiple integration systems with overlapping functionalities. Hence, the core idea is to virtualize a number of heterogeneous physical integration systems by one logical integration system.

Here, we identified the main challenges and explained the conceptual overall architecture. Subsequently, we provided details on specific aspects of that vision and we described a system architecture to realize our vision. However, there are lots of open research aspects and huge optimization potential; hence, further detailed investigation is absolutely required. In conclusion, the major problems in the area of integration processes (the high development effort, the low degree of portability, and the inefficiency) can be overcome by the general concept of *invisible deployment*.

References

1. Boehm, M., Habich, D., Lehner, W., Wloka, U.: Model-driven development of complex and data-intensive integration processes. In: MBSDI (2008)
2. Boehm, M., Wloka, U., Habich, D., Lehner, W.: Model-driven generation and optimization of complex integration processes. In: ICEIS (1) (2008)
3. Dessloch, S., Hernández, M.A., Wisnesky, R., Radwan, A., Zhou, J.: Orchid: Integrating schema mapping and etl. In: ICDE (2008)
4. Albrecht, A., Naumann, F.: Managing etl processes. In: NTII (2008)
5. Stonebraker, M.: Too much middleware. In: SIGMOD Record 31(1) (2002)

6. Boehm, M., Habich, D., Wloka, U., Bittner, J., Lehner, W.: Towards self-optimization of message transformation processes. In: ADBIS (2007)
7. Boehm, M., Habich, D., Lehner, W., Wloka, U.: An advanced transaction model for recovery processing of integration processes. In: ADBIS (2008)
8. Boehm, M., Habich, D., Lehner, W., Wloka, U.: Workload-based optimization of integration processes. In: CIKM (2008)
9. Aulbach, S., Grust, T., Jacobs, D., Kemper, A., Rittinger, J.: Multi-tenant databases for software as a service: schema-mapping techniques. In: SIGMOD (2008)
10. Jacobs, D., Aulbach, S.: Ruminations on multi-tenant databases. In: BTW (2007)
11. Tsai, C.H., Ruan, Y., Sahu, S., Shaikh, A., Shin, K.G.: Virtualization-based techniques for enabling multi-tenant management tools. In: Clemm, A., Granville, L.Z., Stadler, R. (eds.) DSOM 2007. LNCS, vol. 4785, pp. 171–182. Springer, Heidelberg (2007)
12. Shwartz, L., Ayachitula, N., Buco, M.J., Grabarnik, G., Surendra, M., Ward, C., Weinberger, S.: It service provider's multi-customer and multi-tenant environments. In: CEC/EEE (2007)
13. Ramakrishnan, R.: Cloud computing - was thomas watson right after all? In: ICDE (2008)
14. Scheibler, T., Mietzner, R., Leymann, F.: EAI as a Service - Combining the Power of Executable EAI Patterns and SaaS. In: EDOC (2008)
15. Devlin, B., Gray, J., Laing, B., Spix, G.: Scalability terminology: Farms, clones, partitions, packs, racs and raps. CoRR cs.AR/9912010 (1999)

Customizing Enterprise Software as a Service Applications: Back-End Extension in a Multi-tenancy Environment

Jürgen Müller, Jens Krüger, Sebastian Enderlein, Marco Helmich, and Alexander Zeier

Hasso Plattner Institute, Prof.-Dr.-Helmert-Str. 2-3, 14482 Potsdam, Germany
{juergen.mueller,jens.krueger,sebastian.enderlein,
marco.helmich, zeier}@hpi.uni-potsdam.de

Abstract. Since the emerge of Salesforce.com, more and more business applications tend to move towards Software as a Service. In order to target Small and Medium-sized Enterprises, platform providers need to lower their operational costs and establish an ecosystem of partners, customizing their generic solution, to push their products into spot markets. This paper categorizes customization options, identifies cornerstones of a customizable, multi-tenancy aware infrastructure, proposes a framework that encapsulates multi-tenancy, and introduces a technique for partner back-end customizations with regard to a given real-world scenario.

Keywords: Software as a service, Multi-tenancy, Enterprise resource planning, Design.

1 Introduction

During projects we conducted with Small and Medium-sized Enterprises (SMEs), it turned out that the processes implemented there are rather complex and comparable to processes implemented in larger enterprises. However, high up-front and maintenance costs made customized enterprise software unaffordable for Small and Medium-sized Enterprises (SMEs). In order to be able to tackle this market segment, Software as a Service (SaaS) vendors have to significantly lower the total cost of ownership of their products. According to Chong and Carraro [2], this goal can be achieved by leveraging economies of scale efficiently and by reaching a high degree of automation.

As described in Fink and Markovich [7], companies have to take a decision after purchasing an Enterprise Resource Planning (ERP) system [6,15]. Either the company adapts the best practices modeled in the ERP system according to the processes actually performed or vice versa. Standard processes make companies more indistinguishable and with that wipe away potential competitive advantages. Thus, the trend goes towards customized ERP systems which narrow the gap between company-specific business processes and system-embedded best practices.

Enterprise application providers try to push their solutions into the SME market leveraging an adaptable-horizontal distribution strategy [7] that targets many industries with one underlying product. Industry-specific customizations are built ”on top” of the application instead of being an integral part of it. In order to succeed in an adaptable-horizontal

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 66–77, 2009.
© Springer-Verlag Berlin Heidelberg 2009

strategy, [7] propose the establishment of an ecosystem of partners. These partners are then encouraged to develop custom adaptions and extensions for spot markets.

However, the SaaS paradigm poses new challenges to software vendors. Multi-tenancy [2], as one of its key technologies, bears significant complexity to the implementation of an ERP solution. In a multi-tenancy environment, requests from different organizations are served by one application instance which resides on a shared software and hardware infrastructure. This implies that customizations of the code base for one tenant also affect all other tenants on the specific machine. Thus, multi-tenancy eliminates the possibility to customize applications by changing the code. However, multi-tenancy applications support more clients and scale better because they obviate the need for a dedicated machine per customer [2]. According to [9], the scalability of multi-tenancy systems increases with the degree of sharing between tenants. Thus, multi-tenancy reveals a huge potential for cost savings and makes SaaS applications affordable for SMEs in the first place.

In the context of this paper, we aim to address the area of tension between cost efficiency and customizability by illustrating concepts how to incorporate both of these features. Furthermore, we show how the complexity that comes along with multi-tenancy can be hidden from partners. Consequently following [9], we implement an abstraction layer for back-end customization and provide an exemplary partner customization based on a given real-world scenario.

This paper is organized as follows: Section 2 compares our research to related work. Section 3 structures the topic of customization and introduces the show case. Section 4 describes and explains the implementation with regard to the show case. Section 5 concludes in final remarks and outlines future work.

2 Related Work

Ever since, customization of ERP systems is a major concern for purchase decisions of such a system [22,20,18]. [20] propose a decision support framework in which customization opportunities have a huge impact.

[7] point out that flexibility is especially important in the SME market segment and suggests a strategy in which the manufacturer offers a rather generic ERP system and aims to establish an ecosystem of partners. These partners, in turn, see a business case in adapting the generic product in order to meet customer requirements of specific spot markets.

[10], [2], and [3] describe the need for special architectures in order to leverage the benefits of the SaaS model.

Salesforce.com already runs a multi-tenancy-based business application that is, to a certain degree, freely customizable [4]. Salesforce.com has implemented an architecture that pursues a model-driven approach and strongly separates payload and meta data [17].

[9] discuss several multi-tenancy patterns and introduce the idea of an abstraction layer and its key elements. These elements are security isolation, performance isolation, availability isolation, administration isolation, and isolation of customizations. Basic guidelines are given how these key elements could be implemented.

In contrast to customization in multi-tenancy systems on user interface (UI) and business logic level, the database level is well understood and ongoing research proposes patterns how separation between tenants can be implemented efficiently. [3], [11], and [1] give an overview on the separation of tenants on database level and validate their approaches.

3 Modes of Customization

Given the diverse SME market and the resulting high demand of customization [7], partners have different opportunities for customization. In this context, we propose the following set of customization possibilities – desktop integration, UI customization, and back-end customization.

3.1 Desktop Integration

Desktop integration aims to integrate desktop applications with the SaaS application itself. As a matter of fact, desktop integrations do not run in the context of the SaaS application. Since these integrations are installed on every client, they are usually able to provide a better user experience as the standard UI of SaaS applications.

As already indicated, desktop integrations do not customize or extend the functionality range of the SaaS application at all. They focus on enhancing the interaction of desktop applications with the SaaS application to provide a more efficient work place for users.

A layer of business objects serves as a central tier of integration. This layer consists of publicly accessible services that encapsulate the business objects. This way, an object-oriented programming paradigm shall be exposed to partners.

3.2 User Interface Customization

In contrast to desktop integrations, UI customizations are embedded into the front-end of the SaaS ERP application. These widget-like UI components aim to integrate the ERP software with remote services and this way to extend the functionality of the original ERP application. Extending functionality could also mean to automate business processes by consuming remote services.

In order to enable partners to inject arbitrary widgets into the native UI of the business application, a versatile UI concept is needed. In such a scenario three things need to be considered: data from external resources is displayed, this external data shall probably be shared between widgets, and the state of widgets could affect the control flow of the application.

In order to achieve natively embeddable widgets, a container strategy in combination with a screen-wide data exchange can be applied. Foreign widgets are placed into a container in which they are executed (similar to the concept of iFrames).

3.3 Back-End Customization

Back-end customizations are naturally hosted and executed in the back-end of the business application. This allows developers to inject custom business logic into the

business object layer. This business logic can either be a replacement for existing logic or an extension to already existing logic.

Foremost, this kind of customization focuses on automating new business processes or adapting existing ones within the own application. Since business processes usually require human interaction such as the acquisition of necessary data, back-end customizations might imply customizations on UI level. The data gathered on these customized UIs is usually input for custom business logic which specifically supports the implemented business process.

3.4 Problem Statement

As already stated, the scope of this paper is the enablement of back-end customization in ERP systems. Regarding customization, that means adaptions or extensions should only be visible to the one tenant which owns the specific adaption or extension. Since the application instance is shared between all tenants, changes in the code base affect all tenants operating on a shared machine. This raises the strong need to create polymorphic instances of business objects at run-time according to the requesting tenant to enable customization in the first place.

Furthermore, instance composition at run-time implies the presence of a specification for business objects for each tenant. Consequently following this idea, a model-driven approach of executable models is proposed to implement instance composition at run-time.

Since customized business processes might require additional data, persistency is an issue as well. Fortunately, attributes that need to be persisted can be determined at design-time. Therefore, a flexible way to efficiently persist arbitrary data has to be implemented. We propose the introduction of an abstraction layer which separates business objects from the actual persistency in order to hide further complexity from the developer. Additionally, after introducing an abstraction, the persistency methodology is seamlessly interchangeable.

Since a customization can only be executed online, tools and processes have to be provided to support the development process. Also regarding an aspired ecosystem of partners to develop customizations, extensions, or even verticalizations, these partners should be relieved from tasks not related to the actual development. Among these context activities are code logistics, collaboration between multiple developers, testing and deployment, and marketing and sales for customizations.

3.5 Show Case

In order to illustrate the proposed concepts, our considerations are framed by a real-world problem of a company with 300 employees. The company sells folding facades, grew during the last years, now feels the need to implement an ERP system, and is attracted by the SaaS pricing model. But during the implementation, various customization requirements have been identified. Among them is the need for a product configurator to simplify and accelerate offer generation and to automate sales engineering. This tool is going to be exposed to resellers who can enter sales orders for a clearly defined set of products on their own. The work and the material for entered sales orders are thereafter disposed automatically. Unfortunately, the ERP system which is about

to be implemented, does not offer such a feature by default. Therefore, it is subject to customization.

In the described case, the customization covers the introduction of a completely new business object which stores the configuration and the extension of the business object *Linetem* (which is attached to the business object *Opportunity*) by a reference to this new business object.

Therefore, the new business object *Configuration* is created and attached to an *Lineitem*. The folding facades produced by the company have characteristics such as height, width, color, a glass type, and a product type. All these attributes are persisted. Furthermore, a method to calculate the price of the current configuration is required. The new object structure is visualized in Figure 1 using the Fundamental Modeling Concepts [12]. The Fundamental Modeling Concepts provide a framework for the comprehensive description of software-intensive systems.

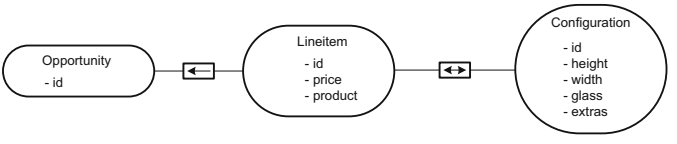


Fig. 1. Custom Business Object Structure

In Figure 2, the original process as well as its improvement are depicted using the Business Process Modeling Notation [21]. In the original process, the reseller gathers sales orders and sends them (via mail or email) to a sales agent of the company. The sales agent validates the sales orders and enters them into the ERP system where they are disposed and further processed. In the improved process, the reseller is able to send the sales orders directly to the ERP system. This process does not require any manual intervention.

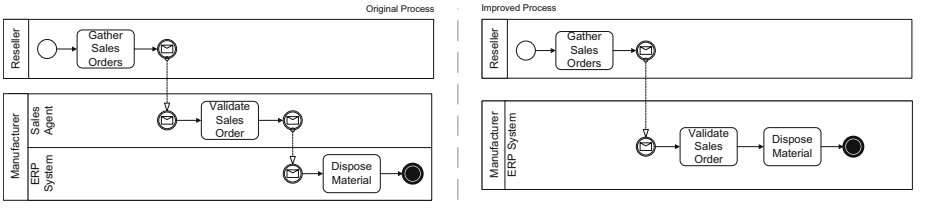


Fig. 2. Process Chart of the original and improved Process

4 Implementation

The goal of this section is to deliver concepts of how to empower partners to inject custom business logic with special regard to customizing and extending business objects (BOs). Business objects (also referred to as domain models [8]) encapsulate business data and associated business logic. We will describe processes, tools, and an infrastructure that affect partners when implementing modifications of BOs. Finally, the proposed

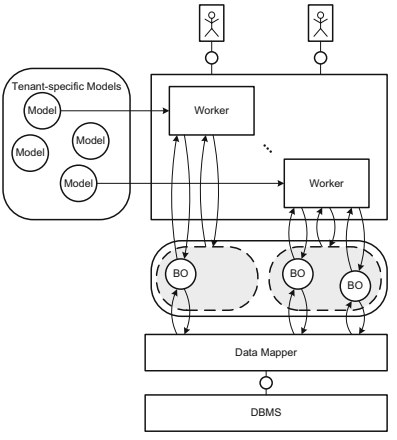


Fig. 3. Block Diagram Back-end Customization

architecture will be explained and proven with an implementation based on the given scenario.

Figure 3 (also visualized using the Fundamental Modeling Concepts [12]) depicts the proposed infrastructure. After a request is received, a worker gets statelessly assigned to it. Depending on a unique tenant identifier, models for the BOs that are about to be used, are requested. Based on these models, run-time objects are created. For each tenant and each BO, exactly one model exists that describes the shape of the particular BO. During and after request processing, BOs are persisted by a data mapper. The data mapper is an independent component which maps BOs to a relational database [8]. The strategy how to store BOs is determined through the implementation of the data mapper. Our model-driven approach and the related persistency concepts are explained in greater detail in Section 4.1.

To implement the proposed architecture, we decided to use Ruby as underlying technology due to its dynamic nature (it offers instance composition by mixing-in Ruby modules at run-time [16]) and rich tool set regarding Web applications.

4.1 Dynamic Instance Composition

As already described, the appearance of multi-tenancy complicates the customization of software systems. Since the system is not dedicated to a single user anymore, changes in the code base affect all tenants running on the system.

From that point of view, customizations have to be stored tenant-specific and separated from the actual code base. Thus, we suggest a model-driven approach in which a model describes the shape of a BO. This description is then interpreted and an instance of this BO is created according to the description. We propose a rather simple but nevertheless expressive grammar, since BOs only consist of attributes, methods, and relationships to other objects (which are a special kind of attributes).

```
<business_object name="Lineitem">
  <properties>
    <property name="id" type="int" access_rights="read" />
    <property name="price" type="float" access_rights="read" />
    ...
  </properties>
  <methods>
    ...
  </methods>
  <extensions />
</business_object>
```

Listing 1: Domain Specific Language Example for *Lineitem* before Extension.

Listing 1 shows an example of the domain specific language [19] that describes the BO *Lineitem*. It contains model elements for business logic, properties, and references to other BOs, which are special properties.

Properties need to be described in a way that they can be easily mapped to data stores (if desired) and visibility as well as access rights can be set. The way properties are persisted is implemented within the data mapper.

The description of methods needs to state where to find the method implementation. In the proposed implementation of the infrastructure, extended business logic is clustered in Ruby modules that are referenced under the ”extensions” element (see Listings 1 and 4) and then mixed into a Ruby instance at run-time. The methods element in our model is primarily used to generate service stubs for easy consumption.

Depending on the persistency strategy, the actual implementation of relationships has to be encapsulated within the data mapper. Relationships within the object-oriented paradigm are implemented through instance variables referencing Ruby objects. Our model reflects this paradigm by treating relationships as properties.

In our show case, we state the need for a new BO *Configuration* and for extending the existing BO *Lineitem*.

The implementation of the BO *Configuration* would look like the declaration of a usual class (see Listing 2).

```
class Configuration < AbstractBo
  def calculate_price
    price = width * height * glass.price ...
  end
end
```

Listing 2: Implementation of *Configuration*.

As described above, extensions to the business logic of existing BOs are achieved through Ruby modules. Listing 3 shows the extension of the implementation of *Lineitem*.

```
module LineitemExtension
  def create_configuration(width, length, glass, extras)
    self.configuration = Configuration.new(width, length, glass,
                                           extras)
  end
end
```

Listing 3: *Lineitem* Extension Sample.

In order to be able to compose an instance of *Lineitem* according to the new shape, the model needs to be modified. Listing 4 illustrates the important changes in the model. It now contains a new property of type *Configuration* and has an extension element which refers to the according implementation.

```
<business_object name="Lineitem">
  <properties>
    ...
    <property name="Configuration" type="Configuration"
                                          access_rights="read" />
  </properties>
  <methods>
    <method  return_type="Configuration"
                         name="create_configuration">
      <param name="width" type="int" />
      ...
    </method>
  </methods>
  <extensions>
    <extension name="LineitemExtension"
                      path="/.../lineitem_extension.rb"/>
  </extensions>
</business_object>
```

Listing 4: Domain Specific Language Example for *Lineitem* after Extension.

The assigned worker (see Figure 3) builds an instance of the BO according to its description. Listing 5 shows the code snippet that is responsible for extending instances of business objects. It is executed in the course of instance creation. Since the actual code base is not changed but only instances of classes are extended, these extensions are neither usable nor visible to other tenants. Moreover, since the type of the instance is not changed, users of each tenant have the illusion that they have a dedicated space with dedicated objects available.

Given the fact that persistency is a major concern in ERP systems, especially regarding dynamically composed instances, according tables in a database need to be available at run-time. Due to the model-driven architecture of our infrastructure, these tables can be generated at design-time. At the point in time a customization is made available, so

called migrations are derived from the business object model and invoked. These migrations are Ruby scripts that create, alter, and delete tables within a database. Attributes specified within the model can now be directly mapped to the according column. This way, the data mapper hides the complexity of persistency (see the data mapper pattern [8]). Our solution implements the Extension Table Layout as described in [1]. In this database layout, associations rely on foreign key mechanisms. Since the data mapper encapsulates the mapping rules and the business object layer is persistency-agnostic, the underlying persistency layout can be changed by re-implementing the data mapper.

```
def initialize_instance
  extensions = BOBuilder.getBOExtensions(object_type, tenant_id)
  extensions.each do |ext|
    # where ext[1] is the path to the extension file
    load ext[1]
    # where ext[0] is the name of the extension
    self.extend eval ext[0]
  end
end
```

Listing 5: Dynamic Instance Composition.

4.2 Partner Context Activities

In order to be able to develop effectively and agile, partners need to be relieved from all context activities and concerns that are not directly connected to their respective mission [13].

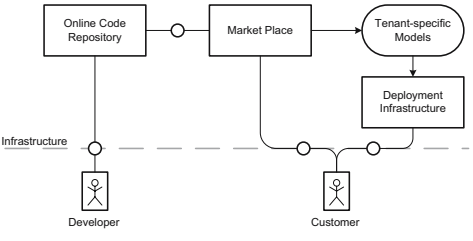


Fig. 4. Block Diagram Developer Framework

Figure 4 depicts facilities offered to developers and customers. Developers have access to their own dedicated hosted code repository [5] which handles version control and code shipping. Furthermore, release shipping can also be conducted through this tool by pulling a special release version. The new release is then added to the hosted market place automatically. After a customer installed an extension, the tenant-specific models are changed according to the specifications of the particular extension. Since the infrastructure is model-driven and the tenant-specific models have changed, the appearance of the customer's application also has changed. The customer is now able to test drive the customized application or further configure it if necessary.

Code Logistics and Version Control. Partners are likely to have prior knowledge in a specific domain and common programming languages and paradigms. It is rather improbable that partners are specialists in hosting with a powerful and scalable infrastructure. Thus, the platform needs to provide all means to manage partner source code, including an integrated testing environment and the ability to collaborate.

We suggest the software vendor to host a code repository that conducts version control and this way enables collaboration between multiple developers. Additionally, problems related to code logistics are solved. After committing code changes, the current version is automatically pulled out of the repository and deployed to the infrastructure. Partners only have to tag a revision as release version. This version is then pulled out of the repository and made available on a market place automatically.

Deployment. We follow a model-driven approach for business logic, leveraging the advantages of object-oriented scripting languages, such as run-time code interpretation and language dynamics. Hence, deploying an extension to a new tenant would simply mean to extend the meta-model of the respective customer organization by linking to the new extension. The potential large number of customers makes a complete automation of the deployment process necessary and the only step a partner should have to take is to activate a specific customization to be available on the application market place (see next paragraph). The actual deployment is performed when customers decide to activate a customization for their organization. The model of affected business objects are extended with according parts so that during the next request the new, extended business objects are instantiated.

Life-Cycle Management. Life-cycle management is a very traditional concern in enterprise software systems but gains new complexity within the SaaS model. Life-cycle management becomes especially interesting when it comes to software updates.

For effectiveness reasons and since a single shared code base encourages the platform provider to shorten the development cycles, it is anticipated to operate all customers on the same version of the application. Since the business object layer is the central point of access, a stable interface needs to be defined. Changes underneath this interface do not affect custom code at all, while changes in the interface inevitably imply changes in every customization which uses that particular piece of code. Therefore, modifications of the interface are strongly discouraged. The life cycle of an SaaS application is rather simple as long as semantics of operations remain untouched and properties and operations are not removed or redefined. Otherwise, changes would require a huge migration effort. Therefore, it is recommended to keep the business object layer rather simple and minimalistic. Additions to it can be conducted by partners or in case of critical additions by the software vendor later on with little effort.

Additionally, partner applications introduce another form of complexity. The proposed model-driven and dynamic nature of the infrastructure also supports the management of their life cycle. The model describes the way business objects are composed. The components, a business object consists of, are referenced at design-time but bound at run-time. This means, changing the reference within a model to a different Ruby module brings up a business object with different functionality. Changes in the version can

be conducted by simply changing the reference within the according model. Assuming the new version is compatible to the old one, the update will happen automatically.

Since business logic is not statically linked to partner applications, their code can be exchanged seamlessly and replaced by a newer, compatible version that is used during the next requests.

5 Conclusions

Providing an ERP platform for Small and Medium-sized Enterprises raises a lot of new questions. Most of them are concerned with the appearance of multi-tenancy. Thus, mastering multi-tenancy is one of the keys to provide an efficient and customizable platform for business applications.

In this paper, we identified different categories of possible customization modes and explored their opportunities. Desktop integrations are rather independent applications which try to integrate the SaaS application with the user's desktop. UI customizations are small widget-like components that are extending the functionality of the SaaS solution by consuming remote services. This type of customization is executed in the direct context of the business application. Back-end Customizations are the most invasive possibility to customize a SaaS application. Here, custom business logic is injected into and executed within the back-end in order to better support special business processes.

Furthermore, we identified two cornerstones of a multi-tenancy aware infrastructure in the context of customization. These are dynamic instance composition and abstraction from the persistency layer. The underlying design and implementation principles were explained on basis of a real-world use case.

Additionally, a framework was proposed that provides partners with an expressive and easy-to-use tool set. This framework is designed to make development of customizations as intuitive as possible to leverage the benefits of a rich partner ecosystem.

Yet untouched topics are the storage and management of business object models as well as the implementation of a data mapper. We plan to conduct further research on these topics as well as on customization on UI level.

References

1. Aulbach, S., Grust, T., Jacobs, D., Kemper, A.: Multi-tenant databases for software as a service: Schema-mapping techniques (2008), http://www-db.in.tum.de/~rittinge/publications/mtdb.pdf
2. Chong, F., Carraro, G.: Architecture strategies for catching the long tail (2006), http://msdn.microsoft.com/en-us/library/aa479069.aspx
3. Chong, F., Gianpaolo, C., Wolter, R.: Multi-tenant data architecture (2006), http://msdn.microsoft.com/en-us/library/aa479086.aspx
4. Coffee, P.: Busting myths of on-demand: Why multi-tenancy matters (2007), http://wiki.apexdevnet.com/images/0/04/MythbustMultiT.PDF
5. CollabNet, I.: subversion.tigris.org (2006), http://subversion.tigris.org/
6. Davenport, T.H.: Putting the enterprise into the enterprise system. Harvard Bus. Rev. 76, 121–131 (1998)

7. Fink, L., Markovich, S.: Generic verticalization strategies in enterprise system markets: An exploratory framework. Journal of Information Technology, 0268–3962 (2008)
8. Fowler, M.: Patterns of Enterprise Application Architecture. Addison-Wesley Longman Publishing Co., Inc., Boston (2002)
9. Guo, C.J., Sun, W., Huang, Y., Wang, Z.H., Gao, B.: A framework for native multi-tenancy application development and management. In: The 9th IEEE International Conference on E-Commerce Technology and the 4th IEEE International Conference on Enterprise Computing, E-Commerce, and E-Services, 2007. CEC/EEE 2007, pp. 551–558 (2007)
10. Hamilton, J.: On designing and deploying internet-scale services. Technical report, Windows Live Services Platform, Microsoft (2007)
11. Jacobs, D., Aulbach, S.: Ruminations on multi-tenant databases. In: Kemper, A., Schöning, H., Rose, T., Jarke, M., Seidl, T., Quix, C., Brochhaus, C. (eds.) BTW. LNI, GI, vol. 103, pp. 514–521 (2007)
12. Knoepfel, A., Groene, B., Tabeling, P.: Fundamental Modeling Concepts: Effective Communication of IT Systems. Wiley, Chichester (2005)
13. Moore, G.A.: Living on the Fault Line. HarperCollins Publishers (2002)
14. Motwani, J., Subramanian, R., Gopalakrishna, P.: Critical factors for successful erp implementation: exploratory findings from four case studies. Comput. Ind. 56(6), 529–544 (2005)
15. Quinn, J.: Intelligent enterprise: a knowledge and service based paradigm for industry. Free Press (1992)
16. rubylang.org: ruby-lang.org (2008), http://www.ruby-lang.org/
17. Salesforce: salesforce.com (2008), http://www.salesforce.com/
18. Somers, T., Nelson, K.: The impact of critical success factors across the stages of enterprise resource planning implementations. In: Hawaii International Conference on System Sciences, vol. 8, p. 8016 (2001)
19. van Deursen, A.v., Klint, P., Visser, J.: Domain-specific languages: An annotated bibliography. SIGPLAN Notices 35, 26–36 (2000)
20. Vilpola, I., Kouri, I., Vaananen-Vainio-Mattila, K.: Rescuing small and medium-sized enterprises from inefficient information systems–a multi-disciplinary method for erp system requirements engineering. In: HICSS 2007: Proceedings of the 40th Annual Hawaii International Conference on System Sciences, Washington, DC, USA, p. 242b. IEEE Computer Society Press, Los Alamitos (2007)
21. Weske, M.: Business Process Management: Concepts, Languages, Architectures, 1st edn. Springer, Heidelberg (2007)
22. Zhang, L., Lee, M.K.O., Zhang, Z., Banerjee, P.: Critical success factors of enterprise resource planning systems implementation success in china. In: HICSS 2003: Proceedings of the 36th Annual Hawaii International Conference on System Sciences (HICSS 2003) - Track 8, Washington, DC, USA, p. 236. IEEE Computer Society, Los Alamitos (2003)

Pattern-Based Refactoring of Legacy Software Systems

Sascha Hunold[1], Björn Krellner[2], Thomas Rauber[1], Thomas Reichel[2], and Gudula Rünger[2]

[1] University of Bayreuth, Germany
{hunold,rauber}@uni-bayreuth.de
[2] Chemnitz University of Technology, Germany,
{bjk,thomr,ruenger}@cs.tu-chemnitz.de

Abstract. Rearchitecturing large software systems becomes more and more complex after years of development and a growing size of the code base. Nonetheless, a constant adaptation of software in production is needed to cope with new requirements. Thus, refactoring legacy code requires tool support to help developers performing this demanding task. Since the code base of legacy software systems is far beyond the size that developers can handle manually we present an approach to perform refactoring tasks automatically. In the pattern-based transformation the abstract syntax tree of a legacy software system is scanned for a particular software pattern. If the pattern is found it is automatically substituted by a target pattern. In particular, we focus on software refactorings to move methods or groups of methods and dependent member variables. The main objective of this refactoring is to reduce the number of dependencies within a software architecture which leads to a less coupled architecture. We demonstrate the effectiveness of our approach in a case study.

Keywords: Pattern-based transformation, Legacy system restructuring, Business software, Class decoupling, Object-oriented metrics.

1 Introduction

The problem of maintaining legacy software is more relevant than ever since many companies are facing the problem of adapting their product lines to new technologies and to short release cycles. During the evolution of software systems new requirements are brought up and old specifications change. In many cases the original software has been built by developers who have left the company years ago. In another scenario, the software architecture has to be reorganized since design decisions have to be adapted by the current developers who very often do not have a complete overview of the entire software. Eick et al. denote this process as *code decay* [6]. The necessary restructuring or rearchitecturing of software systems is a cost-intensive and error-prone task with a high risk of failure when not planned in detail. Some popular software development techniques like *agile software development* or *extreme programming* try to reduce these risks by integrating refactoring and restructuring in the development process [2].

In addition to the actual development process, tool support is required to perform software rearchitecturing tasks especially to minimize risk of errors by using automated

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 78–89, 2009.
© Springer-Verlag Berlin Heidelberg 2009

transformations. Modern integrated development environments (IDEs) support the developer with various transformations (mainly refactorings) or source code generation from user-defined templates (e.g., user interface builders, code completion). Most tools do not enforce a clean separation of concerns when developing an architecture for a specific design pattern, for instance. These tasks have to be realized by the developer on his own.

Rauber and Rünger proposed an incremental transformation process which addresses the problem of restructuring a monolithic business software [14]. This process consists of three steps which have to be traversed to transform the monolithic legacy software into a distributed and modular software system. In the first step (extraction phase), the source code of the legacy system is parsed and transformed into a language independent representation which is called Flexible Software Representation (FSR). The FSR captures all relevant parts of the code structure (e.g., classes or functions) and their dependencies. Furthermore, the source code is annotated to uniquely identify constructs of the programming language in the model. In the second step (transformation), the software is transformed into a high level abstraction layer, preferably using a model driven approach for refactoring. In the last step (generation), the target code is generated using the annotated legacy source code and iteratively applying the defined transformation operations from step 2.

Applying the transformation process mentioned above to real legacy software systems raises several new questions. A very common problem is that the original developer of some code is not known. Thus, it is extremely time-consuming for other developers to figure out what a piece of code is meant to be doing. For this reason, we propose a pattern-based transformation process to improve the software quality automatically. Since quality of source code is hard to measure we rely on software metrics such as code coupling metrics. We consider a software system as improved when software entities are less coupled. Code coupling metrics are based on the number of dependencies between software entities. Therefore, removing dependencies between entities can improve the legacy software since loosely coupled code is easier to modify or to adapt to new requirements.

The proposed automatic transformation process consists of two steps. In the first step, the code of the legacy software system is scanned for predefined patterns to identify the bad smells of software design. This pattern is represented as a graph. The abstract syntax tree (AST) of a legacy program contains the corresponding call graph. The ASTs are scanned in order to find a pattern match. If a pattern is found in the graph, a predefined transformation rule is applied to perform the actual syntax change. It is required that the semantic rules should stay exactly the same. Since this is hard to prove most transformation rules are relatively simple but hard to find by hand.

The contribution of this article is a novel pattern-based transformation process for legacy software systems that can help to automatically remove dependencies between entities. This in turn leads to an improved software architecture. The rest of the paper is organized as follows: Section 2 outlines the incremental transformation process and describes the toolkit TRANSFORMR which implements this process chain. Furthermore, we describe the information which has to be captured within the intermediate representation in order to perform a pattern-based software transformation. Section 3 introduces

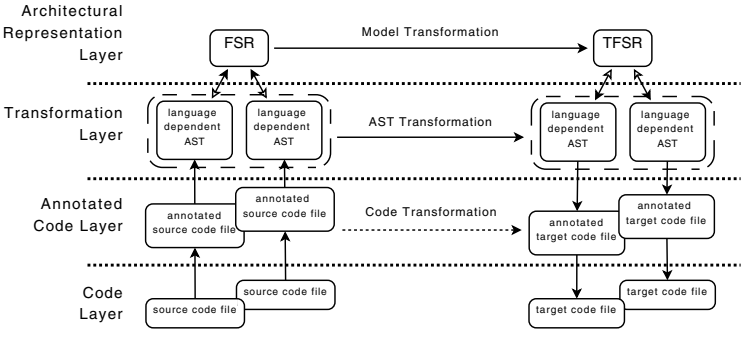


Fig. 1. Abstraction layers of TRANSFORMR. Upwards: model extraction; sidewards: model and code transformation; downwards: code generation.

our automatic approach for software refactoring using pattern-based modifications of ASTs. Its effectiveness is shown by applying the pattern-based refactoring to an example project. Section 5 discusses related work and Section 6 concludes the article.

2 Legacy Software Transformation

2.1 Incremental Software Transformation

Software systems can be classified into different software categories like numerical libraries, operating systems, or business software. Each of these classes requires special strategies and methods in order to perform a software rearchitecturing, e.g., migrating to new operating systems or integrating new technologies. We consider the case of monolithic business software systems. Such a business software consists of a single application with several graphical or text-based user interfaces and a database system. A major goal of our work is to create and implement a software transformation process which helps to decompose a legacy software into modules. A modular description of a software makes it possible to perform all kinds of different transformations, e.g., to port the system to a distributed platform, to integrate new features, or to substitute several modules by more efficient implementations.

The software transformation process is divided into three steps [9]. The first step is the *extraction phase* in which the legacy code is converted into an abstract software model. The structure of the source code is analyzed, e.g., the relationship between classes and methods. Moreover, semantic information (comments, package information) is extracted if possible and attached to the software model. The abstract software model is captured using the flexible software representation. The FSR is an intermediate language to express the structure and the relationships of the source code in a language-independent way. In order to uniquely identify elements of code, e.g., variables or methods, the source code gets annotated in the extraction phase. A major challenge of this step is the categorization of the legacy code into predefined categories such as UI-related code, business logic code, or database-related code. These categories are helpful for later transformations and perception of poorly located functionality. This abstract software model (FSR) enables the software transformation on a higher abstraction level.

The second step is the *transformation phase*. Starting with the FSR, multiple transformations of miscellaneous categories can be applied to the software system. Transformations can vary from simple refactorings (like renaming) to complex ones, like integrating web services. The transformations can be divided into the following categories:

- **basic transformations:** refactorings like rename, move, and create,
- **filter transformations:** to select certain functionality to be kept in the final product,
- **composite transformations:** to relocate functionality onto remote servers, see [9] for more details.

Applying the transformations incrementally leads to the so-called Target FSR (TFSR).

In the last step, the *generation phase*, source code for the target platform is generated from the TFSR. In this step, it is important to generate and to reuse as much code as possible to reduce risks of introducing new bugs.

2.2 TransFormr

The toolkit TRANSFORMR [9] supports the incremental transformation process. The toolkit guides the developer through all phases of the transformation process (extraction, transformation, generation). Figure 1 depicts the abstraction layers which are traversed during the transformation process using the TRANSFORMR toolkit. The source code of the legacy business system is located in the code layer. The annotated code layer contains the code base enriched with TRANSFORMR annotations and forms the basis of the higher abstraction layers. The transition from annotated to executable source code is done by removing all annotation information. The transformation layer is comprised of abstract syntax trees of the annotated source code. These ASTs are then traversed and stored into the flexible software representation which is basically a language-independent description of the language-dependent ASTs. The FSR is located at the top of the abstraction model, combines all information about the software system, and holds references to the source code in order to perform transformation operations.

All transitions between the described layers are performed with a language transformation processor (LTP). We chose TXL [3] as LTP to annotate and to extract the model from the legacy code. It is also used to generate the target code from the model. TRANSFORMR has also been extended to parse comments associated with classes, methods, variables, or statements. This semantic information can be used during the model extraction to separate syntactic and semantic information in the software model.

Most of the transformation operations have to be applied manually by the software architect, i.e., the developer has to select which refactoring operations should be executed. In order to support the architect, TRANSFORMR provides several views on the software, e.g., showing dependencies of a class subset. The visualizations of the software structure, e.g., class, call, or statement dependency diagrams, as well as several software metrics can help observing and evaluating the incremental changes and consequences during the transformation process. We use Coupling Intensity (CINT) and Coupling Dispersion (CDISP) metrics [10] as well as metrics of the following collections: Metrics for Object-Oriented Design (MOOD), Metrics for Object-Oriented Software Engineering (MOOSE), and Quality Metrics for Object-Oriented Design (QMOOD), summarized in [13]. All are variably adapted to our intermediate software model.

All FSR elements contain links to their extracted semantic information (e.g., comments, categorization), for the visualization and transformation. In the generation stage, the information is exported as comments according to comment guidelines, i.e., it is inserted at the appropriate places in the generated source code.

3 Pattern-Based Moving of MemberGroups

The support for detecting and moving some functionality within legacy code remains a key issue for decoupling software modules. The TRANSFORMR toolkit addresses this problem by providing a pattern-based search to detect separated concerns in classes and several ways to move method code and member variables between classes. Since moving static or global methods and variables around can be easily done, the present work mainly focuses on relocating member methods or member variables.

In the following, two major types of move operations of member methods are considered:

- **Delegation** copies the header and body of the method into another class and replaces the old method's body by a call to the new target method. All references to the old method stay unmodified.
- **Explicit Moving** means that in addition to the moving of program code all references to the method are replaced by an appropriate call to the new method in the target class.

Both move operations have in common that the signature of the moved method has to be altered if public methods of the source class are accessed within the method. In that case, a new parameter is added to the method which passes a reference to the source class. The move operations cannot be performed if the method to be moved accesses private members of the source class because they are not accessible from the target class. In that case, the visibility of the accessed members has to be changed to overcome this problem.

Delegation is often used if the old class tends to be too complex even if the method is semantically located in the proper class. To reduce complexity and inner class coupling the functionality is moved into a newly created class which should not be visible to other classes in the system as they still call the original method.

If a method is moved explicitly, the functionality of the method should be inserted into a class which fits best. The main limitation of this operation is that a reference to the target class is needed wherever the moved method was called on the source class previously.

A variation of the explicit moving is to make the method a member function of one of its parameters. This special moving can be applied if the coupling of the method to the parameter class is bigger than to the own class. The following example (Figure 2) demonstrates this case. The method `Source:m(Target)` is tightly coupled with class `Target` because it calls only methods of `Target`. Moving `m(Target)` into class `Target` is obvious in this case. Additionally, the parameter `t` is eliminated and the reference `s.m(t)` is replaced with `t.m()` in class `Another`. This procedure is applicable in all cases and is automatically done by IDEs, like Eclipse or NetBeans, for trivial cases

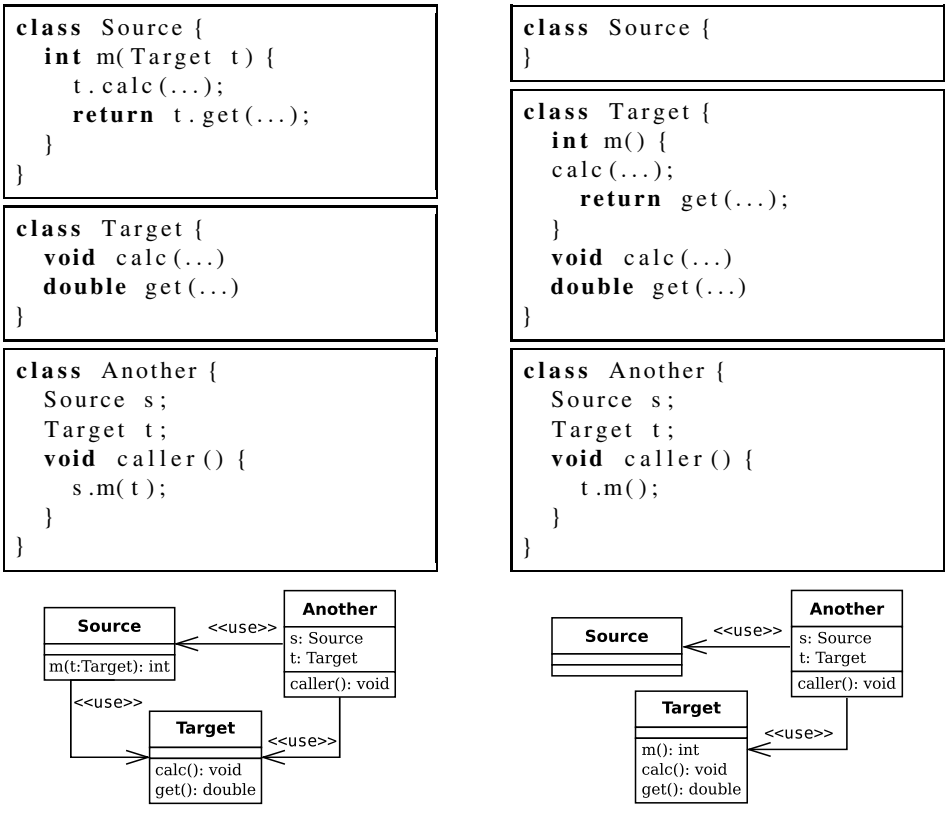


Fig. 2. Example of moving the method `m()` into the parameter class `Target`. Left: initial class model, right: class model after moving `m()`.

like the example above. If the method which should be moved contains dependencies to other members, like the use of a private member variable, the refactoring engines of the IDEs fail or break encapsulation by rising the visibility of private variables (see Figure 3).

To address these problems, we propose a refactoring strategy which helps to detect groups of methods and member variables which have the same concerns and to move these groups between classes. We introduce the term *MemberGroup* to denote such a group. A *MemberGroup* consists of exactly one public method that can access other private members (methods or member variables) of the same class which are not used by other methods. *MemberGroups* often occur if a method's task is split into sub-tasks that are implemented by a couple of private methods and use private member variables of the parent class. If we want to move the public method of the *MemberGroup*, it is useful to move all members (the public method and all accessed private members) of the *MemberGroup* to capture the whole concern. In this paper, we primarily consider members which are not inherited from superclasses, overridden by child classes, or implement methods of an interface. The preconditions for moving those members are not affected by the architectural constraints of the class design.

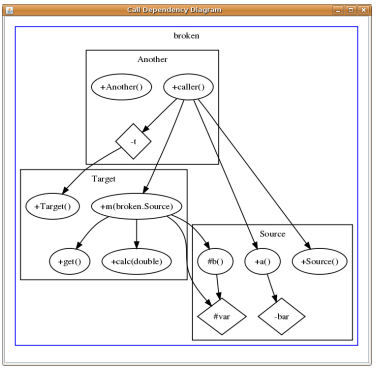
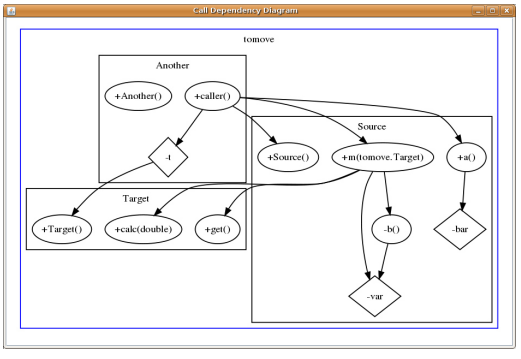


Fig. 3. Broken encapsulation of class `Source` (outward edges of `Target` to former private members of `Source`)



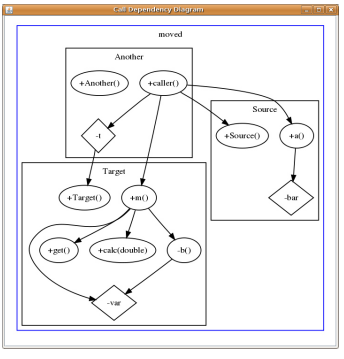


Fig. 4. Left: Example of strong class coupling which can be reduced by moving *MemberGroup* (`m(Target), b(), var`). Right: Class dependencies after moving the *MemberGroup*.

Based on the description, we propose a strategy that supports the pattern-based transformation process.

1. Build a software model (FSR) of the legacy system with the toolkit TRANSFORMR.
2. Search for *MemberGroups* inside of the classes with graph patterns on the FSR of the software.
3. For each *MemberGroup*: Present possibilities to move the *MemberGroup* and indicators for each one.
   (a) Move into parameter class: Class coupling of *MemberGroup* is bigger to parameter class than to original class.
   (b) Delegate *MemberGroup*: Could be used if the public method implements or extends an existing one.
   (c) Explicit move: The class coupling to another class is bigger than to the original class.
4. Validate preconditions and apply the transformations on the software model.

To move a *MemberGroup* with delegation (3b) or explicitly (3c), manual interaction is necessary to obtain the reference to the target class in each class in which the *MemberGroup* is used. In case of (3a) the strategy can be performed fully automated and used to reduce the Class Coupling in order to improve the understandability and maintainability of a legacy system (see Section 4).

Code metrics are used to indicate the usefulness of moving a *MemberGroup* to some target class. The metrics are based on the number of outward and inward edges in the call dependency graph. The Coupling Intensity (CINT) [10] metric is defined as the number of distinct method calls from a given method (outward edges). The Coupling Dispersion (CDISP) is defined as the number of classes in which a method is called (number of inward edges) divided by CINT.

Based on the ideas of [10] we introduce indicators to find target classes for *MemberGroups*.

- Move into a parameter class C_p if the *MemberGroup* has more than one edge to C_p. If more than one class is available, use the class with the highest number of edges from the *MemberGroup* to C_p.
- Move the *MemberGroup* into the class with the highest CDISP value and:
  - Use delegation if the *MemberGroup* is semantically correct placed but in a too complex or oversized class or if the public method of the *MemberGroup* implements or extends an existing one;
  - Otherwise use explicit moving.

Automatic moving of a *MemberGroup* is not always possible, e.g., if the public method of the *MemberGroup* implements an interface method. In such a scenario, the developer can be supported by presenting call dependency diagrams with depicted *MemberGroups* and indicators for target classes in order to perform the code change manually.

4 Experimental Analysis

In this section, we propose an example of moving a *MemberGroup* into a parameter class applying the strategy described in the previous section. Figure 4 depicts an example of strongly coupled classes on the left hand side. The coupling can be removed by moving the *MemberGroup* (`m(Target), b(), var`) to the parameter class `Target`. An indicator for moving the *MemberGroup*, is the number of outward edges from the *MemberGroup* to other classes (two edges to class `Target` vs. no edges to `Source` and `Another`).

When we apply the strategy described in Section 3 to the example in Figure 4, the graph search finds the *MemberGroup* pattern (`m(Target), b(), var`) in class `Source`. In order to find possible target classes into which the *MemberGroup* can be moved several metrics are calculated, summarized in Table 1. Based on the calculated metrics and the number of inward and outward edges of the *MemberGroup*, we suggest `Target` as target class because of CINT(`Target`)=2 and since `Target` is a parameter of method `m(Target)`.

The result of moving the *MemberGroup* into `Target` is shown in Figure 4 (right). The major improvement after moving the *MemberGroup* is the decoupling of the classes

Table 1. Code metrics for the *MemberGroup* (`m(Target), b(), var`)

Class	CINT	CDISP	Inward Edges
`Target`	2	0	0
`Another`	0	-	1

`Source` and `Target`. This coupling improvement can be measured with the class coupling metric (CC) of the MOOD metrics set [1]. Class coupling metric is defined as the ratio of the sum of the class pair couplings $c(C_i,\ C_j)$ and the overall number of class pairs in a system of n classes. The coupling $c(C_i, C_j) = 1$ if there is a dependency between C_i and C_j (method call or variable), otherwise 0.

$$\text{CC} = \frac{\sum_{i=1}^{n} \sum_{j=1, i \neq j}^{n} c(C_i, C_j)}{n^2 - n}$$

The use of the metric in the example results in the following improvement in the class coupling metric:

	CC
before moving *MemberGroup*	$\frac{3}{6} = 0.5$
after moving *MemberGroup*	$\frac{2}{6} = \mathbf{0.33}$

For legacy applications a lower coupling is desired since a higher coupling increases complexity, reduces encapsulation and potential reuse, and limits understandability and maintainability [1].

In a separate study we decomposed the source code of the Apache Jakarta project JMeter[1] (809 classes, 70 kLOC) into *MemberGroups*. A total of 206 of these *MemberGroups* with exactly one public method and at least one private member were found. Due to the fact that the detected *MemberGroups* have to match certain constraints the proposed class transformation could not be applied. Sample constraints of such a transformation are: (a) all classes which hold a reference to the source class also have to hold a reference to the target class, or (b) all methods within a *MemberGroup* must not be part of an interface. Even though no target class could be found for this particular case, the study shows that TRANSFORMR can be used to decompose a software project into *MemberGroups* and that it checks the necessary constraints to apply particular transformations.

5 Related Work

The use of patterns is a fundamental principle of software engineering. In contrast to our work, in which we try to exploit design patterns of the legacy software, it is also suitable to use patterns when building a software architecture from scratch. A pattern-based approach for the development of a software architecture is presented in [5]. The main

[1] http://jakarta.apache.org/jmeter/

idea of this work is to break down the software design problem into several subproblems and to apply a software pattern (called pattern frames) to solve the subproblems. This makes it possible to change certain design decisions during the evolution of the software by, e.g., instantiating a different design pattern for the implementation of a subproblem.

Fowler et al. introduced many refactorings and design patterns for object-oriented languages as solutions for common mistakes in code style in order to make the software easier to understand and cheaper to modify [8]. The proposed manual changes in the software design are supported by tests to verify the correctness of the software. Other work describes the need for automatic support during refactoring and restructuring tasks but also state the limits and drawbacks of full automated restructuring (e.g., untrustworthy comments, meaningless identifier names) [12].

As in our approach, metrics can be used to suggest possible target classes to move functionality [7]. In contrast to our work, the authors move only single methods and present an Eclipse plug-in to detect code that suggests refactorings (bad smells) in Java projects. Based on a distance metric between classes and methods the plug-in suggests methods to move if the distance to another class is lower than to the original class. The authors applied their approach to two projects mentioned in [8], and conclude that the plug-in was able to detect a great amount of bad smells which Fowler et al. suggested for these projects as well. Thus, distance or coupling metrics can help to detect misplaced methods [11].

A reengineering methodology for a given software is proposed in [4]. It consists of three steps: create a source code representation (Program Representation Graph (PRG)), transform this representation, and generate target code. The approach defines orthogonal code categories with a concern (user interface (UI), business logic, or data), roles (definition, action, and validation), and controls as connectors between concerns. The categorization process is mainly driven by the categorization of a set of base classes into the concerns (e.g., GUI library classes) followed by a categorization of variables, attributes, and procedures which use the already categorized set. Based on the categorized PRG the authors outline a general move method transformation to detect methods with different concerns in order to separate UI code from data access code.

6 Conclusions

In this article, we have presented a novel approach to perform automated transformations of legacy software. The main goal of the proposed procedure is to support developers by changing the software architecture of a legacy system. We focus on obtaining a better separation of concerns by removing class dependencies automatically. The transformation procedure works as follows: The developer defines or selects a legacy software pattern. This pattern represents a dependency graph. The legacy software is searched for occurrences of this pattern. If the legacy pattern is found a pre-defined target pattern is inserted. As example pattern the *MemberGroup* move pattern was introduced. A *MemberGroup* consists of the publicly accessible class method and its dependent private member methods and class members. An algorithm is presented which finds *MemberGroups* in a legacy system and suggests appropriate target classes based on code coupling metrics. An experimental evaluation shows by example how legacy

code can be improved if the proposed transformation method is applied. To justify this pattern-based refactoring process, it is also shown that the legacy patterns considered here can actually be detected in real world software systems.

In future work, we plan to extend the *MemberGroup* move pattern to capture more functional concerns in legacy software. One possible enhancement could be to weaken the restrictions on the visibility of the group members, e.g., dependent methods could also have a public modifier. Another possible pattern could be to identify multiple dependent public methods, e.g., getter and setter functions.

Acknowledgements. The transformation approach described in this article as well as the associated toolkit are part of the results of the joint research project called TransBS funded by the German Federal Ministry of Education and Research.

References

1. Abreu, F., Brito, R.: Object-Oriented Software Engineering: Measuring and Controlling the Development Process. In: Proc. of the 4th Int. Conf. on Software Quality (ASQC), McLean, VA, USA (1994)
2. Beck, K., Andres, C.: Extreme Programming Explained: Embrace Change, 2nd edn. Addison-Wesley Professional, Reading (2004)
3. Cordy, J.R.: Source transformation, analysis and generation in TXL. In: Proc. of the 2006 ACM SIGPLAN Symp. on Partial Evaluation and Semantics-based Program Manipulation (PEPM 2006), New York, NY, USA, pp. 1–11(2006)
4. Correia, R., Matos, C., El-Ramly, M., Heckel, R., Koutsoukos, G., Andrade, L.: Software Reengineering at the Architectural Level: Transformation of Legacy Systems. Technical report, University of Leicester (2006)
5. Côté, I., Heisel, M., Wentzlaff, I.: Pattern-based Exploration of Design Alternatives for the Evolution of Software Architectures. Int. Journal of Cooperative Information Systems (December 2007) (Special Issue of the Best Papers of the ECSA 2007)
6. Eick, S.G., Graves, T.L., Karr, A.F., Marron, J.S., Mockus, A.: Does Code Decay? Assessing the Evidence from Change Management Data. IEEE Transactions on Software Engineering 27(1), 1–12 (2001)
7. Fokaefs, M., Tsantalis, N., Chatzigeorgiou, A.: JDeodorant: Identification and Removal of Feature Envy Bad Smells. In: Proc. of the 23rd IEEE Int. Conf. on Software Maintenance (ICSM 2007), Paris, France, pp. 519–520 (October 2007)
8. Fowler, M., Beck, K., Brant, J., Opdyke, W., Roberts, D.: Refactoring: Improving the Design of Existing Code. Addison-Wesley Professional, Massachusetts (1999)
9. Hunold, S., Korch, M., Krellner, B., Rauber, T., Reichel, T., Rünger, G.: Transformation of Legacy Software into Client/Server Applications through Pattern-Based Rearchitecturing. In: Proc. of the 32nd IEEE Int. Computer Software and Applications Conf (COMPSAC 2008), Turku, Finland, pp. 303–310 (2008)
10. Lanza, M., Marinescu, R., Ducasse, S.: Object-Oriented Metrics in Practice. Springer, New York (2006)
11. Mäntylä, M.V., Lassenius, C.: Drivers for software refactoring decisions. In: Proc. of the 2006 ACM/IEEE Int. Symp. on Empirical Software Engineering (ISESE 2006), New York, NY, USA, pp. 297–306 (2006)

12. Mens, T., Tourwé, T.: A Survey of Software Refactoring. IEEE Transactions on Software Engineering 30(2), 126–139 (2004)
13. Portugal, L., Baroni, L.: Formal Definition of Object-Oriented Design Metrics. Master's thesis, Ecole des Mines de Nantes, France; Universidade Nova de Lisboa, Portugal (2002)
14. Rauber, T., Rünger, G.: Transformation of Legacy Business Software into Client-Server Architectures. In: Proc. of the 9th Int. Conf. on Enterprise Information Systems, Funchal, Madeira, Portugal (2007)

A Natural and Multi-layered Approach to Detect Changes in Tree-Based Textual Documents

Angelo Di Iorio[1], Michele Schirinzi[1], Fabio Vitali[1], and Carlo Marchetti[2,3]

[1] Dept. of Computer Science, University of Bologna
Mura Anteo Zamboni 7, 40127, Bologna, Italy
[2] Dip. di Informatica e Sistemistica, University of Rome "La Sapienza"
Via Ariosto 22, Rome, Italy
[3] Senato Della Repubblica Italiana, Palazzo Madama, Rome, Italy

Abstract. Several efficient and very powerful algorithms exist for detecting changes in tree-based textual documents, such as those encoded in XML. An important aspect is still underestimated in their design and implementation: the quality of the output, in terms of readability, clearness and accuracy for human users. Such requirement is particularly relevant when diff-ing literary documents, such as books, articles, reviews, acts, and so on. This paper introduces the concept of 'naturalness' in diff-ing tree-based textual documents, and discusses a new extensible set of changes which can and should be detected. A naturalness-based algorithm is presented, as well as its application for diff-ing XML-encoded legislative documents. The algorithm, called JNDiff, proved to detect significantly better matchings (since new operations are recognized) and to be very efficient.

Keywords: XML Diff-ing, Changes detection, Naturalness, Data management.

1 Introduction

The way users create, store and edit some kind of data has been changing in the recent years. The boundaries between structured data (where information is organized in tuples and records) and textual documents (where information is encoded as a stream of text) have progressively been fading. A leading role in such a process has been played by XML, with its strong accent on the coexistence between *human*- and *machine*-readable documents. Users are often not only interested in the current version of XML-encoded documents but also in their history and changes. The automatic detection of differences among them is then destined to become more and more important.

Although XML is used to encode both literary documents and database dumps, there is '*something different*' from diff-ing an XML-encoded literary document and a XML-encoded database. Two observations support our idea, from different perspectives. First, the fact that the output of a *diff* on literary documents needs to be as much as possible faithful to the output of a 'manual' *diff*. Such a property, which is undoubtedly true in any context, is much more relevant for literary resources because they are primarily meant to be read by humans. The second and more important point is about the editing model of literary documents: they are usually modified according to some patterns and rules different from those adopted in changing databases. This behavior can be then exploited to produce *high-quality and natural* outputs.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 90–101, 2009.
© Springer-Verlag Berlin Heidelberg 2009

This paper proposes a new dimension to evaluate, design and implement algorithms for diff-ing XML documents: the *naturalness*. In a sense, the naturalness indicates the capability of an algorithm to identify those changes which would be identified by a manual approach. This work address various aspects related to the notion of naturalness, at different levels. In particular, we:

- discuss an extensible set of natural changes which an algorithm specialized for literary documents can and should detect.
- present an algorithm, called JNDiff, able to detect most of them. We also stress on the modularity and high configurability of JNDiff.
- describe a Java system based on these ideas. We also present a case-study on detecting changes between XML-encoded legislative bills, and focus on benefits of such natural approach for their editing/publishing workflow.

The rest of the paper is then structured as follows. Section 2 describes related works; section 3 introduces the concept of naturalness. JNDiff is analyzed in section 4, while section 5 briefly describes JNMerge, a tool to interpret and re-build the output of JNDiff. The evaluation of both these tools is presented in section 6, while section 7 is about the case-study on legislative documents.

2 Related Works

A variety of tools which calculate differences among XML documents exists. Generally, comparison methods are divided in two groups: systems that operate on generic text content (such as GNUDiff [2]) and systems specifically designed for XML content. We define here a further breakdown into specific methods to compare XML documents, by distinguishing: (i) *diff* working on XML documents that express database dump, and (ii) *diff* working on XML documents, that express literary documents.

Cobéna et al. [7] proposed XyDiff to detect changes between two ordered XML trees T_1 and T_2. This algorithm uses a two-pass technique: XyDiff starts with matching the nodes by matching their ID attributes; next, it computes a signature and a weight for each node of both trees in a bottom-up traversal. A top-down phase - repeated in order for each subtree - completes the algorithm. XyDiff algorithm achieves $O(nlogn)$ complexity in execution time and generates fairly good results in many cases. However, it cannot guarantee any form of optimal or near optimal result because of the greedy rules used in the algorithm.

X-diff [11] detects changes on parsed unordered labeled tree of XML. X-diff finds the equivalent second level sub-trees and compares the nodes using the structural information denoted as signature. In order to detect move operations i.e., if a node is moved from position i in the old tree to position j in the new tree an ordered tree is needed. Zhang and Shasha proposed a fast algorithm to find the minimum cost editing distance between two ordered labeled trees [12]. Given two ordered trees T_1 and T_2, in which each node has an associated label, their algorithm finds an optimal edit script in time $O(|T_1| \times |T_2| \times mindepth(T1), leaves(T1) \times mindepth(T2), leaves(T2))$, which is the best known result for the general tree-to-tree correction problem.

There are very few algorithms customized for literary XML documents. The AT&T Internet Difference Engine [3] [4] uses an internal module to determine the differences

between two HTML pages. That module treats two HTML pages as two sequences of tokens (a token is either a sentence-breaking markup or a sentence) and uses a weighted LCS algorithm [8] to find the best matching between the two sequences. DeltaXML [5] [6] developed by Mosnell provides a plug-in solution for detecting and displaying changes between two versions of a XML literary document. They represent changes in a merged delta file by adding additional attributes to the original XML file.

3 Naturalness in Diff-ing Literary Documents

There is one aspect still underestimated in the design and implementation of XML diff algorithms: the quality of the output, in terms of readability, clearness and accuracy for human users. This aspect is particularly relevant for literary documents such as books, review, reports, acts, when encoded in XML. The point is that most of the existing diff-ing algorithm capture *changes on the trees representing these documents* rather than *changes on the documents themselves*.

Let us explain such issue with a simple example. Consider an author who merges two paragraphs in a document and deletes a fragment of the second one.

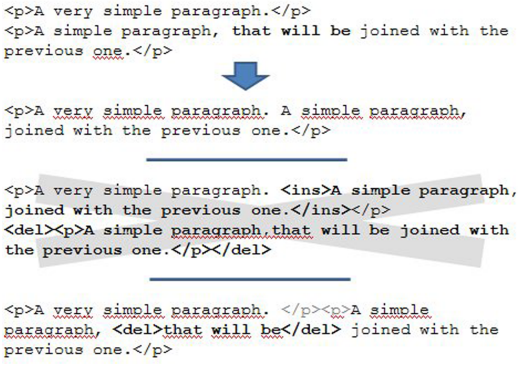


Fig. 1. Merging two paragraphs

Fig. 1 shows such a case in HTML (input files are followed by two possible diff-ing outputs). In the best case, all the algorithms we are aware of detect a couple of independent changes: the deletion of the whole second paragraphs, and the insertion of some words in the first one. Such output is technically correct but it would undoubtedly be more useful to detect a paragraphs' re-structuring. Syntactical details are not relevant here, but it is important to understand how to improve quality when dealing with literary documents.

The authors of such documents do not strictly operate on the underpinning tree structure of the documents themselves. They basically edit, delete, change and move higher-level structures. For instance, they may insert/remove nodes along a path (wrapping text with a bold or italic effect), restructure text-nodes into hierarchical sub-trees (dividing an act into sub-acts, paragraphs, etc.), flatten a structured paragraph into a text nodes (removing formatting), rename elements (translating labels of a document), and so on.

Our idea is then to study 'meaningful and common' operations/changes which reflect what authors actually do on literary documents. We then introduce the notion of '*naturalness*' of a diff-ing algorithm. The '*naturalness*' indicates the '*capability of an algorithm to identify* ***real*** *changes, i.e. changes which would be identified by a manual approach*'.

In order to define the set of 'natural' operations, we propose a layered approach: it consists of exploiting regularities and patterns of editing processes, and reformulating high-level changes as combinations of atomic ones.

Basically, the existing algorithms identify four operations: (i) insertion of a subtree, (ii) deletion of a subtree, (iii) update of a node and (iv) moves of a subtree. Although these operations fully express changes in the documents' trees, they hardly express higher-level changes mentioned so far (merging paragraphs, adding formatting, refactoring text blocks, and so on). What we propose is then to extend such traditional model into an open one, able to capture all those very common editing actions.

3.1 A New Set of Natural Operations

Paradoxically, the first relevant aspect of the set of operations we propose is their incompleteness. Our current model is the result of a deep analysis of the most common editing processes. On the other hand, we expect to discover new meaningful changes, tailored for specific contexts, specific classes of documents and specific editing patterns. A preliminary set of operations is listed below:

Insertion/Deletion. Inevitably our model comprises the operations of insertion and deletion, as any textual diff-ing algorithm does. They represent the 'bricks' upon which other complex operations are built. Moreover we define either the insertion/deletion of a subtree of of an element along a path.

Move. The movement of a subtree is another natural change which should be detected. Moving fragments of a document is in fact a very common operation (suffice it to mention the cut&paste operation we do thousand and thousand of times) on literary documents, already detected by most existing algorithms.

Downgrade. The *downgrade* operation occurs when adding nodes along a path, as shown in figure 2. Actually two subtrees are downgraded, after adding an intermediate element. The downgrade operation is a very good example of natural diff on literary documents. While in a database context such operation is quite uncommon (records can be moved or updated, but their subcomponents are hardly pushed down in a hierarchical structure), they are very common when dealing with hierarchical texts and managing mixed content-models, containers and sub-containers, logical document structures.

Upgrade. As expected the opposite *upgrade* operations occurs when a node is removed along a path. Figure 2 can be read from right to left in order to picture an upgrade operation. There is actually a deletion of element that does not involve its whole subtree but only the element itself.

Even upgrades are very common when editing literary documents, since authors use to change, polish and re-organize (sub-)structures of a document, without working on whole subtrees but on 'connections' between them.

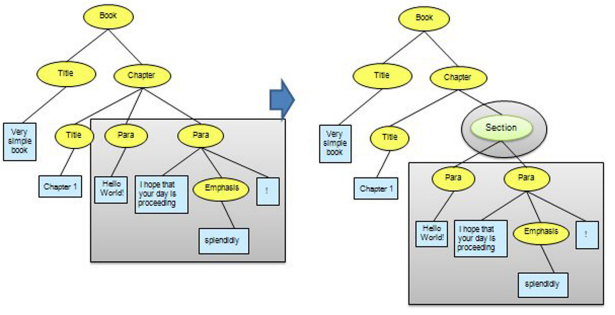


Fig. 2. The *downgrade* operation

Refactoring. We define the *refactoring* operation as a structural modification of "contiguous" blocks in a textual content. Refactoring is very frequent operation in document editing. Suffice it to mention how many times we divide a paragraph or we insert and remove emphasis on fragments.

.

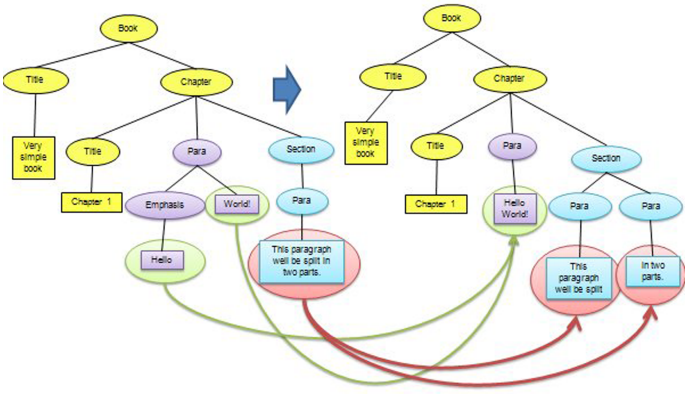


Fig. 3. The *refactoring* operation

Figure 3 shows an example: a single text node (right-bottom of the tree) is splitted in two, while composite text node is created by deleting an in-line element. The refactoring is a high-level operation that aggregates some elementary changes, considered as a whole action. Some of them are: (i) *Join(X,Y)* (merging content of X and Y nodes), (ii) *Split(X,n)* (dividing the X node content at the n offset into two splitted text nodes), (iii)*InsNode* (deleting a new single node), (iv) *DelNode* (deleting a single node). What is important is the logical consistency of a document before and after the application of a refactoring operation, as a combination of all these changes.

Element Update. A very common operation on XML documents is the *update* of an element, a change which does not involve its content-model (and name) but only its attributes. Insertion/deletion of an attribute, and modification of its value are possible

sub-steps of such complex change. For instance, style and formatting adjustments on literary documents may correspond to such updates.

Text Node Update. One of the most common editing changes is the *text-node update*, i.e. insertion/deletion of substrings in a text. We propose to aggregate such update in a complex operation of elementary changes (insertion/deletion of a substring of a given length in a given position). Detecting this type of changes is very important in literary documents because it allows us to perform a very exhaustive text changes analysis until single word level.

4 An Optimized Algorithm for Natural (XML) Diff-ing: JNDiff

The set of changes described in the previous section are independent from any diff-ing algorithm. A first goal of our work, in fact, was investigating a new diff-ing approach for literary tree-based documents, and defining a set of natural operations to be recognized. The second and more important step is designing an actual algorithm able to detect those changes. Since any complex operation is actually a combination of atomic ones, we could have extended existing algorithms with an interpretation-phase able to rearrange the output in terms of higher-level changes. Instead, we have implemented a **native naturalness-based** algorithm using specific data structures and rules to **directly** detect natural changes. We called it JNDiff, as it is implemented in Java.

Intuitively, we tried to 'simulate' in JNDiff our experience in *diffing* literary documents. What we humans do is trying to understand which are the relationships between parts of the two input documents. We found different relationships with an iterative process: we usually first identify those parts which remain unchanged, in order to have a sort of *pivot* around which other changes are detected and classified. Then, we identify as 'moved' those parts which do not change but are in a different position, or as 'updated' those parts which have been slightly modified but do not change their position (similarly, we can identify as 'upgraded' or 'downgraded' those unmodified parts which have been pushed/pulled downward/upward in the tree document structure).

JNDiff adopts a similar approach: it is a modular algorithm which first detects a set of relationships between the documents' parts (basically a list of insertions/deletions) and iteratively refines them, through cascade phases. Each phase is in charge of detecting a specific class of changes.

The modularity is one of the most important and innovative aspects of JNDiff. It allows us to customize the algorithm on the basis of users' preferences and needs. Although the current implementation of JNDiff works much better with literary documents, the algorithm can be easily specialized for different applications domains. For instance, we can obtain much better results when diff-ing database dumps by deactivating modules for upgrades/downgrades detection, since these operations are very uncommon in that context. Similarly, other configurations can be set up for different scenarios. Moreover, new modules can be implemented able to detect new operations (as we said before, the set of changes we propose here is a partial list meant to be extended and polished) and easily activated. The algorithm is then extremely powerful and flexible.

In the rest of the paper we discuss the current configuration/implementation of JNDiff which have been tailored for naturalness-based diff-ing. It consists of five independent phases:

Phase 0: Linearization. The preliminary phase, called *linearization*, is mandatory and independent from the set of changes we need to detect. For each input document, it creates a 'smart' data structure, called *VTree*, which makes it easy and fast to identify and compare documents' elements and subtrees. Basically, a *VTree* is an array of records built with a pre-order depth-first visit of a document. Each record represents a node and contains: a hash-value which identifies that node (and its attributes), a hash-value which identifies the whole subtree rooted in that node (derived from the hash-values of its children and itself), a pointer to that node in the document, and other application-dependent information we dot not describe here, due to space limits. Such data structure plays an essential role for JNDiff, since it allows to compare elements by simply comparing integer numbers, *in constant time* (in particular, whole subtrees can be matched by matching two integers). Note also that the construction of a VTree is linear on the number of nodes, since JNDiff basically visits twice a tree and properly synthesizes meaningful hash-values.

Phase 1: Partitioning. The *Partitioning* phase consists of finding unchanged parts of the two documents. As expected, such a match greatly benefits from the VTree data structures. The search of those subtrees, in fact, becomes a search of a LCS (Longest Common Subsequence) between two arrays. Note also that a subtree corresponds to a continuous interval of a VTree, because of the pre-order depth-first visit, and many comparisons can be skipped. JNDiff finds a LCS as an ordered concatenation of LCSS (Longest Common Continuous Subsequence). Actually other algorithms for LCS could be used, such as Myers's one [10].

At this stage, unchanged parts are identified and connected by a single type of *matching* relation. The rest of the nodes are then considered *inserted*, if they only are in the second document, or *deleted*, if they only are the first one.

Phase 2: Text Updates Detection. The phase 2 is the first optional step of JNDiff, and detects text nodes updates. Two criteria are followed to do that: *principle of locality* and *similarity threshold*. They respectively state that two text-nodes are considered 'updated' if (i) they lay between two subtrees matched in phase 1 (intuitively, they belong to the same document part) and (ii) the amount of text changes crosses a threshold passed as a parameter. These tests are in fact designed to simulate the 'natural' behavior described at the beginning of the section (updated nodes do not change their position and do slightly change their content).

The parametrization of the similarity threshold is another key aspect to be remarked: once again, JNDiff is highly configurable and different classes of updates can be easily detected by changing that threshold.

Phase 3: Moves Detection. The phase 3 is a further optional step which detects moves. Basically, JNDiff measures the distance between matched subtrees (by counting matching partitions and nodes between them) and classifies as 'moved' nodes whose distance

is under a given threshold. As happens for updates, such a solution has a twofold goal: (i) it simulates manual changes detection (nodes are very often moved within a limited range in a document) and (ii) it is highly configurable and can be adapted to very different contexts.

Phase 4: Matches Expansion. The last phase we have implemented is called *matches expansion* and propagates changes bottom-up, in order to improve the quality of the output. It is then optional but highly recommended to have much more natural results. Intuitively, JNDiff 'goes up' from leaves to the root and fixes some 'imperfections' in the interpretation of insertions/deletions. Some elements in fact are still recognized as inserted/deleted, even if they are actually unchanged. The reason is that their (VTree) hash-values have changed because something have changed in their descendants (for instance, the *book* or *chapter* elements of the sample). What JNDiff does is then removing those false positives among insertions/deletions and polishing the output. Moreover, by analyzing some specific VTree fields, it detects updates of elements' attributes.

Advanced Phases. The phases discussed so far were designed to obtain natural diffing outputs. In fact, most of the changes defined in section 3.1 are currently detected by JNDiff: inserted/deleted subtrees belong to the residual class of unmatched nodes, downgraded/upgraded nodes have a deleted/inserted element among their ancestors, moves and text-updates are recognized by phases 2 and 3, and elements' updates are recognized during phase 4.

What is alike important is the modularity of JNDiff. In fact, we plan to work on new modules to detect more sophisticated changes. In particular, we aim at detecting combined operations such as refactoring (as described in section 3.1) or identifying nodes which have been simultaneously moved (or upgrade/downgrade) and slightly updated. The current engine, in fact, does not detect such tangled changes (only one match-relation is found, according to the order and parameters of the phases).

5 Expressing Detected Changes: JNMerge and JNApply

Currently JNDiff is implemented as a Java application, running on common web servers or executable by command line. Both these interfaces allows users to specify parameters useful to customize the algorithm. The output of JNDiff is an XML file which lists changes to be applied on the input document A in order to obtain the input document B. Such output (Δ) expresses very fine-grained and diversified differences but it is quite complex and application-dependent. Syntactical details are not relevant here (moreover they will be probably changed). What is important is the fact that JNDiff is not enough to clearly show users how documents changed. We then implemented a related tool which generates a document $A + \Delta$ that clearly expresses those changes. We called it JNMerge.

What JNMerge does is scanning such output and embedding changes in the document A. For each detected operation, it adds appropriate attributes and other markup to the original document. Basically, JNMerge uses indexes and pointers of a *VTree* in order to access and modify the document.

By exploiting that information, the original document A is progressively transformed into A + Δ. Let us discuss a simple case: when adding a node along a path (*InsertNode*) that node is added to its new parent, all children of its parent are 'adopted' by the new child, and all references are accordingly updated. Similarly, each operation implies a well-defined set of modifications.

As expected, such re-building is a very complex process. The document JNMerge deals with, in fact, is significantly different from the document JNDiff evaluated, since information about previous changes have already been embedded and offsets and positions are broken. However JNMerge rebuilds a correct and meaningful A + Δ. Details of such process are out of the scope of this paper, whose main topic is JNDiff. Syntactical details of JNMerge markup are also not relevant now, since they can (and will) be easily changed.

We also implemented JNApply, a tool that directly generates the second document B taking in input A and Δ. It is an optimization which could have been coded by extending JNMerge with transformations on the embedded *delta*.

6 Computational Complexity of JNDiff and JNMerge

The most important goal of JNDiff was improving the quality of the output. A first application confirmed that very good results can be achieved, as discussed in the next section case-study. On the other hand, it is important to evaluate the algorithm in terms of computational costs. In fact, JNDiff is admittedly less efficient than other algorithms, just because it aims at maximizing 'naturalness' first of all. By considering the overall result, however, we conclude JNDiff is a very good trade-off between naturalness and complexity.

Defining the complexity of JNDiff is quite difficult, since that measure strongly depends on which phases are actually executed. That is why we briefly discuss each phase separately. We will express complexity in terms of four parameters: n (nodes number of document A), m (nodes number of document B), p (leaves number of document A), q (leaves number of document B).

The VTree linearization (section 4) is realized by simply visiting the DOM trees, so it costs $\Theta(n + m)$. The partitioning-phase (section 4) consists of finding a LCS in the linearized VTrees. The best known algorithm is Myers'one [10] that attains to find a LCS in time $O((n + m)D)$, where D is the length of a shortest edit script. JNDiff uses a different approach in order to find a LCCS, a LCS with longest contiguous substrings. As mentioned before such approach captures the largest and most 'natural' subtrees, especially for literary documents. It finds a LCCS in $O(n \times m)$ but it has a $\Omega(1)$ lower bound. In fact, he internal structure of VTrees makes it easy and fast to compare identical subtrees.

Thus, JNDiff works very well on documents with slight differences, while it has worse results when processing very different documents. However we expect that literary documents have much more unmodified nodes and subtrees, than changed ones. When editing literary documents, in fact, users tend to modify limited parts of them by adding/removing text fragments, re-organizing parts, adding/removing nodes along paths, and so on.

Similar considerations can be applied to the optional phases of JNDiff. For instance, the complexity of text-updates-detection (section 4) is $O(p \times q \times f(p,q))$, where f is the function that calculates the similarity between nodes p e q (f costs $O(length(p) \times length(q))$ because uses our LCCS algorithm but can be fastened by using Meyrs's one, to the detriment of naturalness). Similarly, a move-detection phase (section 4) costs $O(n \times m)$ when all nodes need to be scanned. In the same way, the complexity of the match-expansion phase (section 4) is $O(min(p,q) \times log(n))$ since JNDiff has to rise the tree for $log(n)$ nodes.

The total complexity of JNDiff is then $O(n \times m)$. In practice, since each phase works only on nodes not yet connected and the majority is connected after initial phases, computational costs progressively decreases at each interaction.

7 A Practical Application: Detecting Changes in Legislative Documents

As laid down by the Italian Constitution, the Italian Parliament consists of two Houses: the Senate of the Republic and the Chamber of Deputies. According to the principle of full bicameralism, these two houses perform identical functions. The law-making function is thus performed jointly by the two Houses: a legislative bill becomes an act only after it has been passed by both Houses of Parliament *in the same exact wording*. As consequence, it is important to provide Senators and Deputies with effective means for analyzing the modifications applied to legislative bills in a House after it has been modified in the other. The main tool provided by Senate employees to Senators for this purpose are documents containing the so-called "*Testo a Fronte*" (TAF in the following, which literally means *forehead text*). A TAF document is a two columns page layout document that permits to represent the differences between the original version of the legislative document and the modified version. In a TAF document, the left column is used to represent the original version of a legislative bill, while the right column represent the modified one. However, in order to put in evidence modifications, the two columns don't merely list the aligned content of the two versions: they rather *re-present* these versions according to well-defined *TAF presentation rules*, which permit a quick identification of the differences to readers (some real-world TAF examples are shown on our demo web site *http://twprojects.cs.unibo.it:8080/taf/*).

A TAF presentation rule is determined on the basis of the applied modification, which can span either the overall structure of the document (e.g. articles, commas) or simply a portion of the text (e.g. rephrasing of a comma). For the sake of synthesis, we describe a simplified version of some of these rules:

1. if some words within a comma (or its sub-parts) are deleted in the new version, then the text on the *left* column is printed in **bold face** and the right column is left unmodified,
2. if some words within a comma (or its sub-parts) are inserted in the new version, then the text on the *right* column is printed in **bold face** and the left column is left unmodified,
3. if an article is suppressed in the new version, then all the article text on the *left* column is printed in **bold face** while the one on the *right* is substituted with a blank paragraph starting with "*Soppresso.*" (i.e., suppressed),

4. if an article is left unchanged in the new version, then the text on the *left* column remains identical while the one on the *right* is substituted with a blank paragraph starting with "*Identico.*" (i.e., identical).

Note that rules 1 and 2 deal with modifications to text only, rule 3, and 4 deal with modification to the overall legislative bill structure. Let us also remark that these examples of TAF presentation rules are actually oversimplified: for the sake of synthesis, they omit several important details on alignments and punctuation, among the others.

The process for producing TAF documents summarized above is intrinsically error-prone and can be very costly in absence of appropriate tools (considering the length of some legislative documents). It is very useful to automatize such a process of evaluating and printing TAF documents in Senate. Indeed, an application permitting to approximate a good-quality TAF document (namely, *TAF-1.0*) is already under evaluation in some drafting Offices of the Italian Senate. TAF-1.0 is obtained by engineering and integrating together an implementation of JNDiff, augmented with a sophisticated set of XSL transformation to obtain a printable and human readable TAF document. The overall application is delivered as a Java servlet. In order to apply JNDiff to legislative documents, these must be valid XMLs. TAF-1.0 obtains the XML versions of a legislative bill using the *XMLeges marker* component [1], a syntactical parser enabling to transform the flat text of a bill in a structured XML document compliant with the NIR (*Norme in Rete*) Italian Standard for the markup of legislative acts [9]. NIR XML documents have a fine grained markup that enables JNDiff to produce very meaningful sets of revealed differences. By implementing TAF presentations rules, XSL transformations of TAF-1.0 then completes the job: they are used to transform the JNDiff output and the NIR XML bill versions in an HTML TAF document, which is provided to users of drafting offices for further refinements, integrations, final formatting, and printing using normal word processors. We recently deployed a first version of TAF-1.0 and results are more than promising. Our application is available at *http://twprojects.cs.unibo.it:8080/taf/*, along with some demonstration documents.

8 Conclusions

Our research on *naturalness* and JNDiff is not completed yet. However, the current implementation of JNDiff is a reliable, modular and open-source application available at *http://jndiff.sourceforge.net/*. The related tool JNMerge is also coded in Java and available at the same web site.

Our next step will be a further investigation of the set of changes proposed here, in order to (i) implement modules which detect still unsupported changes and (ii) identify more complex high-level operations. We also plan to perform deeper tests on thresholds and parameters passed to JNDiff. Further evaluations of computational costs and resource consuming will be investigated as well.

References

1. Agnoloni, T., Francesconi, E., Spinosa, P.: xmLeges Editor, an OpenSource visual XML editor for supporting Legal National Standards. In: Proceedings of V Legislative XML Workshop, Florence, Italy (2007)

2. Eggert, P.: Free Software Foundation: GNU Diff (2006), http://www.gnu.org/software/diffutils/diffutils.html
3. Ball, T., Douglis, F.: Tracking and viewing changes on the web. In: 1996 USENIX Annual Technical Conference (1996)
4. Chen, Y.F., Douglis, F., Ball, T., Koutsofios, E.: The at&t internet difference engine: Tracking and viewing changes on the web. World Wide Web 1(1), 27–44 (1998)
5. Fontaine, R.L.: A delta format for xml: identifying changes in xml files and representing the changes in xml. In: XML Europe 2001 (May 2001)
6. Fontaine, R.L.: Xml files: a new approach providing intelligent merge of xml data sets. In: XML Europe 2002 (May 2002)
7. Marian, A., Cobena, G., Abiteboul, S.: Detecting changes in xml documents. In: The 18th International Conference on Data Engineering, February 2002, pp. 493–504 (2002)
8. Hirschberg, D.S.: Algorithm for the longest common subsequence problem. Journal of the ACM 24(4), 664–675 (1977)
9. Lupo, C., Aini, F.: Norme in rete (1999), http://www.normeinrete.it/
10. Myers, E.W.: An o(nd) difference algorithm and its variations. Algorithmica 1(2), 251–266 (1986)
11. Cai, J., Wang, Y., DeWitt, D.: X-diff: an effective change detection algorithm for xml documents. Technical Report, University of Wisconsin (2001)
12. Zhang, K., Shasha, D.: Simple fast algorithms for the editing distance between trees and related problems. SIAM Journal of Computing 18(6), 1245–1262 (1989)

CrimsonHex: A Service Oriented Repository of Specialised Learning Objects

José Paulo Leal[1] and Ricardo Queirós[2]

[1] CRACS & DCC-FCUP, University of Porto, Portugal
[2] CRACS & DI-ESEIG/IPP, Porto, Portugal
zp@dcc.fc.up.pt, ricardo.queiros@eu.ipp.pt

Abstract. The corner stone of the interoperability of eLearning systems is the standard definition of learning objects. Nevertheless, for some domains this standard is insufficient to fully describe all the assets, especially when they are used as input for other eLearning services. On the other hand, a standard definition of learning objects in not enough to ensure interoperability among eLearning systems; they must also use a standard API to exchange learning objects. This paper presents the design and implementation of a service oriented repository of learning objects called crimsonHex. This repository is fully compliant with the existing interoperability standards and supports new definitions of learning objects for specialized domains. We illustrate this feature with the definition of programming problems as learning objects and its validation by the repository. This repository is also prepared to store usage data on learning objects to tailor the presentation order and adapt it to learner profiles.

Keywords: eLearning, Repositories, SOA, Interoperability.

1 Introduction

Component oriented systems are predominant in most of eLearning platforms. Despite their success, they have also been target of criticism: their tools are too general and they are difficult to integrate with other eLearning systems [1]. These issues led to a new generation of service oriented eLearning platforms, easier to integrate with other systems. This paper focuses the design and implementation of crimsonHex, a service oriented repository of specialized learning objects (LO). It provides standard compliant repository services to a broad range of eLearning systems, exposing its functions using two alternative web services flavours. The definition of LOs can be customized to the requirements of these systems. To illustrate this customization we document the process of extending generic LOs to a specific learning domain – programming exercises.

The extended definition of LOs to programming problems is being used in a European research project called EduJudge. This project aims to integrate a collection of problems created for programming contests into an effective educational environment. This project includes three types of services:

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 102–113, 2009.
© Springer-Verlag Berlin Heidelberg 2009

- Learning Objects Repository (LOR) to store the exercises and to retrieve those
- suited to a particular learner profile;
- Evaluation Engine (EE) to automatic evaluate and grade the students attempt to solve the exercises;
- Learning Management System (LMS) to manage the presentation of exercises to learners.

The remainder of this paper is organized as follows: Section 2 traces the evolution of eLearning systems with emphasis on the existing repositories. In the following section we extend the generic definition of a LO as a programming problem. Then, we present the architecture of the repository and highlight its components, functions and communication model. The next section, we focus on the main facets of its implementation: storage, validation, interface and security. In Section 6 we describe the tests and evaluation of the repository. Finally, we conclude with a summary of the main contributions of this work and a perspective of future research.

2 State of Art

The evolution of eLearning systems comprises the last two decades. In the "first generation", eLearning systems had a monolithic architecture and were used on a specific learning domain [1]. Gradually, these systems evolved and became independent from a particular domain, incorporating tools that can be effectively reused in several scenarios. Different kinds of component based eLearning systems targeted to a specific aspect of eLearning, such as student or course management. There are several acronyms trying to differentiate between these types of eLearning systems. Nevertheless, the trend in eLearning systems is integration therefore most of them evolved to the same set of standard features and many of these acronyms are used as synonyms. The most usual designation of such systems is the LMS (e.g. Moodle, Sakai, and WebCT).

This "second generation" allows the sharing of learning objects and learner information. In this phase, some standards emerge, namely, IMS Content Packaging (IMS CP), Sharable Content Object Reference Model (SCORM) and IEEE Learning Object Metadata (IEEE LOM) that brought interoperability and content sharing to eLearning. Despite the advantages of these systems and standards, some criticism arose for several reasons, such as: focus on content, lack of support to response to specific needs and difficult to integrate with other eLearning systems.

These issues triggered a new generation of eLearning platforms based on services that can be integrated in different scenarios. This new approach provides the basis for a Service Oriented Architecture (SOA) [2]. In the last few years there have been initiatives to adapt SOA to eLearning, such as the eLearning Framework (ELF) and the IMS Abstract Framework. These initiatives contributed with the identification service usage models and a categorisation of genres of services for eLearning [3]. Some of these services are related with a key system in an eLearning platform – the repository.

A repository of learning objects can be defined as a 'system that stores electronic objects and meta-data about those objects' [4]. The need for this kind of repositories is growing as more educators are eager to use digital educational contents and more of

it is available. One of the best examples is the repository Merlot (Multimedia Educational Resource for Learning and Online Teaching). The repository provides pointers to online learning materials and includes a search engine. The Jorum Team made a comprehensive survey [5] of the existing repositories and noticed that most of these systems do not store actual learning objects. They just store meta-data describing LOs, including pointers to their locations on the Web, and sometimes these pointers are dangling. Although some of these repositories list a large number of pointers to LOs, they have few instances in any category, such as programming problems. Last but not least, the LOs listed in these repositories must be manually imported into a LMS. An evaluation engine cannot query the repository and automatically import the LO it needs. In summary, most of the current repositories are specialized search engines of LOs and not adequate for interact with other eLearning systems, such as, feeding an automatic evaluation engine.

Based in other surveys [4] the users are concerned with issues that are not completely addressed by the existing systems, such as interoperability. Some major interoperability efforts [6] were made in eLearning, such as, NSDL, POOL, ELENA/Edutella, EduSource and IMS Digital Repositories (IMS DRI). The IMS DRI specification was created by the IMS Global Learning Consortium (IMS GLC) and provides a functional architecture and reference model for repository interoperability. The IMS DRI provides recommendations for common repository functions, namely the submission, search and download of LOs. It recommends the use of web services to expose the repository functions based on the Simple Object Access Protocol (SOAP) protocol, defined by W3C. Despite the SOAP recommendation, other web service interfaces could be used, such as, Representational State Transfer (REST) [7].

Besides the interoperability features of the repository its necessary to look to the current standards that describes learning objects. As we said before, the actual standards are quite generic and not adequate to specific domains, such as the definition of programming problems. The most widely used standard for LO is the IMS CP. This content packaging format uses an XML manifest file wrapped with other resources inside a zip file. The manifest includes the IEEE LOM standard to describe the learning resources included in the package. However, LOM was not specifically designed to accommodate the requirements of automatic evaluation of programming problems. For instance, there is no way to assert the role of specific resources, such as test cases or solutions. Fortunately, LOM was designed to be straightforward to extend it. Next, we enumerate four ways that have been used [8] to extend the LOM model:

- combining the LOM elements with elements from other specifications;
- defining extensions to the LOM elements while preserving its set of categories;
- simplifying LOM, reducing the number of LOM elements and the choices they present;
- extending and reducing simultaneously the number of LOM elements.

Following this extension philosophy, the IMS GLC upgraded the Question & Test Interoperability (QTI) specification. QTI describes a data model for questions and test data and, unlike in its previous versions, extends the LOM with its own meta-data vocabulary. QTI was designed for questions with a set of pre-defined answers, such as multiple choice, multiple response, fill-in-the-blanks and short text questions. It

supports also long text answers but the specification of their evaluation is outside the scope of the QTI. Although long text answers could be used to write the program's source code, there is no way to specify how it should be compiled and executed, which test data should be used and how it should be graded. For these reasons we consider that QTI is not adequate for automatic evaluation of programming exercises, although it may be supported for sake of compatibility with some LMS. Recently, IMS GLC proposed the IMS Common Cartridge that bundles the previous specifications and its main goal is to organize and distribute digital learning content.

3 Specialised Learning Objects

We defined programming problems as learning objects based on the IMS CP. An IMS CP learning object assembles resources and meta-data into a distribution medium, in our case a file archive in zip format, with its content described in a file named imsmanifest.xml in the root level. The manifest contains four sections: meta-data, organizations, resources and sub-manifests. The main sections are meta-data, which includes a description of the package, and resources, containing a list of references to other files in the archive (resources) and dependency among them.

Meta-data information in the manifest file usually follows the IEEE LOM schema, although other schemata can be used. These meta-data elements can be inserted in any section of the IMS CP manifest. In our case, the meta-data that cannot be conveniently

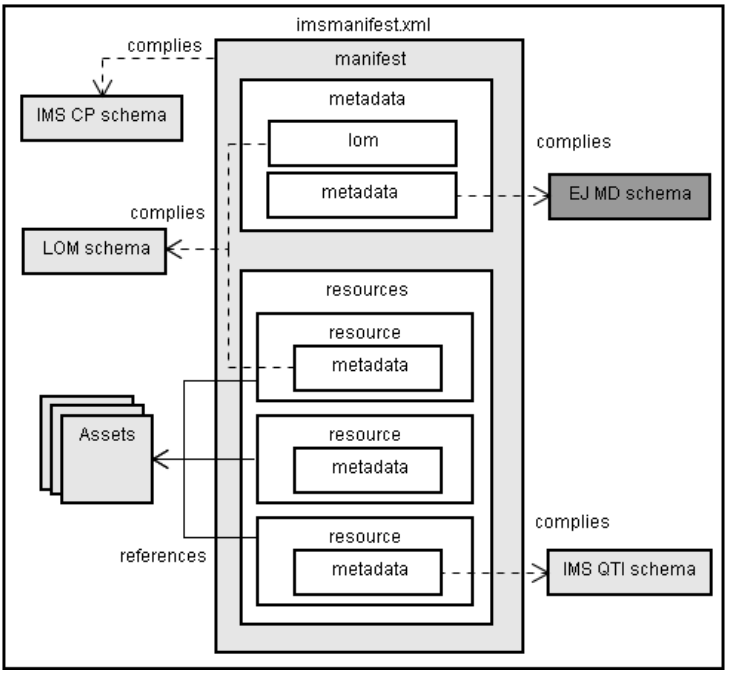


Fig. 1. Structure of a programming problem as a learning object

represented using LOM is encoded in elements of a new schema - EJ MD - and included only in the meta-data section of the IMS CP. This section is the proper place to describe relationships among resources, as those needed for automatic evaluation and lacking in the IEEE LOM. The compound schema can be viewed as a new application profile that combines meta-data elements selected from several schemata. This approach is similar to the SCORM 1.2 application profile that extends IMS CP with more sophisticated sequencing and Contents-to-LMS communication. The structure of the archive, acting as distribution medium and containing the programming problem as a LO, is depicted in Figure 1.

The archive contains several files represented in the diagram as grey rectangles. The manifest is an XML file and its elements' structure is represented by white rectangles. Different elements of the manifest comply with different schemata packaged in the same archive, as represented by the dashed arrows: the manifest root element complies with the IMS CP schema; elements in the metadata section may comply either with IEEE LOM or with EJ MD schemas; metadata elements within resources may comply either with IEEE LOM or IMS QTI. Resource elements in the manifest file reference assets packaged in the archive, as represented in solid arrows.

4 Architecture

In this section, we present the architecture of the crimsonHex repository described by the UML component diagram shown in Figure 2. Using the **API crimsonHex**, the repository exposes a core set of functions that can be efficiently implemented by a simple and stable component. All other features are relegated to auxiliary components, connected to the central component using this API. Other eLearning systems can be plugged into the repository using also this API.

4.1 Components

In the design of crimsonHex we set some initial requirements, in particular, to be simple and efficient. Simplicity is the best way to promote the reliability and efficiency of the repository. In fact, the core operations of the repository are uploading and downloading LO - ZIP archives - which are inherently simple operations that can be implemented almost directly over the transport protocol. Other features may need a more elaborate implementation but do not require the same reliability and efficiency of the core features. The architecture of crimsonHex repository is divided in three main components:

- **The Core** exposes the main features of the repository, both to external services, such as the LMS and the EE, and to internal components - the Web Manager and the Importer;
- **The Web Manager** allows the creation, revision, versioning, uploading/ downloading of LOs and related meta-data, enforcing compliance with controlled vocabularies;
- **The Importer** populates the repository with existing legacy repositories. In the remainder we focus on the Core component, more precisely, its functions, communication model and implementation.

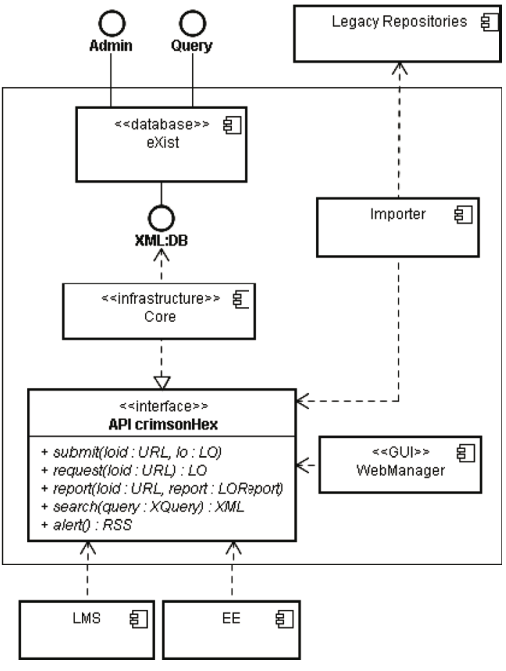


Fig. 2. Components diagram of the repository

4.2 Functions

The Core component of the crimsonHex repository provides a minimal set of operations exposed as web services and based in the IMS DRI specification. The main functions are the following.

The **Register/Reserve** function requests a unique ID from the repository. We separated this function from Submit/Store in order to allow the inclusion of the ID in the meta-data of the LO itself. This ID is an URL that must be used for submitting a LO. The producer may use this URL as an ID with the guarantee of its uniqueness and the advantage of being a network location from where the LO can be downloaded.

The **Submit/Store** function copies a LO to a repository and makes it available for future access. This operation receives as argument an IMS CP with the EJ MD extension and an URL generated by the Register/Reserve function with a location/ identification in the repository. This operation validates the LO conformity to the IMS Package Conformance and stores the package in the internal database;

The **Search/Expose** function enables the eLearning systems to query the repository using the XQuery language, as recommended by the IMS DRI. This approach gives more flexibility to the client systems to perform any queries supported by the repository's data. To write queries in XQuery the programmers of the client systems need to know the repository's database schema. These queries are based on both the content of the LO manifest and the LOs' usage reports, and can combine the two document types. The client developer needs also to know that the database is structured in collections. A collection is a kind of a folder containing several

resources and also other folders. From the XQuery point of view the database is a collection of manifest files. For each manifest file there is a nested collection containing the usage reports. As an example of a simple search, suppose we want to find all title elements in the LO collection with an easy difficulty level.

```
declare namespace imsmd = "http://...";
for     $p in //imsmd:lom
where   contains($p//imsmd:difficulty,easy)
return  $p//imsmd:title//text()
```

The previous example displays a FLWOR ("For, Let, Where, Order by, Return") expression based in XQuery language to locate all such elements. This approach is used in SOAP requests. For REST requests we can simple write in a browser the URL: `http://host/crimsonHex?difficulty=easy`. In both approaches the result is a set of strings; alternatively, it can be a XML document. In this case it is possible to format the result using an XSLT (Extensible Stylesheet Language Transformation) file. For frequent queries it's possible to compile and cache them as XQuery procedures.

The **Report/Store** function associates a usage report to an existing LO. This function is invoked by the LMS to submit a final report, summarizing the use of a LO by a single student. This report includes both general data on the student's attempt to solve the programming exercise (e.g. data, number of evaluations, success) and particular data on the student's characteristics (e.g. gender, age, instructional level). With this data, the LMS will be able to dynamically generate presentation orders based on previous uses of LO, instead of using fixed presentation orders. This function is an extension of the IMS DRI.

The **Alert/Expose** function notifies users of changes in the state of the repository using a Really Simple Syndication (RSS) feed. With this option a user can have up-to-date information through a feed reader.

4.3 Communication Model

The communication model of the repository defines the interaction between the repository and the other eLearning systems. The model is composed by a set of core functions, most of them, exposed in the previous section. The figure 3 shows an UML diagram to illustrate the sequence of core functions invocations from these eLearning systems to repositories.

The life cycle of a LO starts with the reserve of an identification and the submission of a LO to the repository. Next, the LO is available for searching and delivering to other eLearning systems. Then, the learner in the LMS could use the LO and submit it sending an attempt of the problem solution to the EE. Based in the feedback the learner could repeat the process. In the end, the LMS sends a report of the LO usage data back to the repository. This DRI extension will be, in our view, the basis for a next generation of LMS with the capability to adjust the order of presentation of the programming exercises in accordance with the needs of a particular student.

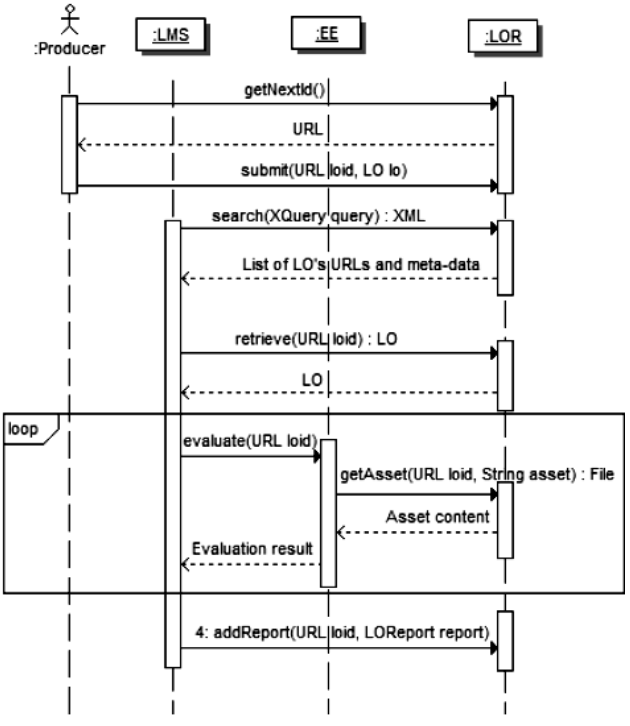


Fig. 3. Communication between the repository and the other eLearning systems

5 Implementation

In this section we detail the design and implementation of the Core component of crimsonHex on the Tomcat servlet container.

Reliability and efficiency were our main concern when designing the Core. The best way to achieve them is through the simplicity. These are the main design goal that guided us in the development of the four main facets of the Core - storage, validation, interface and security - analysed in the following subsections.

5.1 Storage

Searching LOs in the repository is based on queries on their XML manifests. Since manifests are XML documents with complex schemata we paid particular attention to databases systems with XML support: XML enabled relational databases and Native XML Databases (NXD).

XML enabled relational databases are traditional databases with XML import/export features. They do not store internally data in XML format hence they do not support querying using XQuery. Since queries in this standard are a DRI recommendation this type of storage is not a valid option. In contrast, NXD uses the XML document as fundamental unit of (logical) storage, making it more suitable for

data schemata difficult to fit in the relational model. Moreover, using XML documents as storage units enables the following standards:

- XPath for simple queries on document or collections of documents;
- XQuery for queries requiring transformational scaffolding;
- SOAP, REST, WebDAV, XmlRpc and Atom for application interface;
- XML:DB API (or XAPI) as a standard interface to access XML datastores.
- XSLT to transform documents or query-results retrieved from the database.

We analysed several open source NXD, including SEDNA, OZONE, XIndice and eXist, Only eXist implements the complete list of the features enumerated above, which led us to select it as the storage component of crimsonHex. It has also two important features [9] worth mentioning: support for collections, to structure the database in groups of related documents and automatic indexes to speed up the database access**.**

5.2 Validation

The crimsonHex is a repository of specialized learning objects. To support this multi typed content the repository must have a flexible LO validation feature. The eXist NXD supports implicit validation on insertion of XML documents in the database but this feature could not be used for several reasons: LO are not XML documents (are ZIP files containing an XML manifest); manifest validation may involve many XML Schema Definition (XSD) files that are not efficiently handled by eXist; and manifest validation may combine XSD and Schematron validation and this last is not fully supported by eXist.

All LOs stored in crimsonHex must comply with the IMS Package Conformance that specifies it structure and content. This standard also requires the XSD validation of their manifests. For particular domains it is possible to configure specialized validations in crimsonHex by supplying a Java class implementing a specific interface. These validations extend those of the IMS Package Conformance and may introduce new schemata, even using different type definition languages, such as Schematron.

Validations are configured per collection of documents. Thus, different types of specialized LO may coexist in a single instance of crimsonHex. As mentioned before, IMS CP main schema imports many other schemata (more than 30) that according to the IMS Package Conformance must be downloaded from the Internet. This requirement has a huge impact on the performance of the submit function. To accelerate this function we implemented a cache. A newly stored schema has a time to live of 1 hour. Outdated schemata are reloaded from their original Internet location using a conditional HTTP request that downloads it only if it has effectively changed.

5.3 Interface

To comply with standards, the IMS DRI recommends the implementation of core functions as web services. We chose to implement two distinct flavours of web services: SOAP and REST. SOAP web services are usually action oriented, mainly when used in Remote Procedure Call (RPC) mode and implemented by an off the shelf SOAP engine such as Axis.

Table 1. Core functions of the repository

Function	SOAP	REST
Reserve	URL getNextId()	GET /nextId > URL
Submit	submit(URL loid, LO lo)	PUT URL < LO
Request	LO retrieve(URL loid)	GET *URL* > *LO*
Search	XML search(XQuery query)	POST /query < *XQUERY* > *XML*
Report	Report(URL loid,LOReport rep)	PUT *URL*/report < *LOREPORT*
Alert	RSS getUpdates()	GET /rss > RSS

The web services based on the REST style are object (resource) oriented and implemented directly over the HTTP protocol, using, for example, Java servlets, mostly to put and get resources, such as LOs and usage data. The reason to implement two distinct web service flavours is to promote the use of the repository by adjusting to different architectural styles. The repository functions are summarized in Table 1. Each function is associated with the corresponding operations in both SOAP and REST web services interfaces.

5.4 Security

Following the design principles of simplicity and efficiency we decided to avoid the management of users and access control in the Core. This decision does not preclude the security of this component since we can control these features in the communication layer. Since both web services flavours use HTTP as transport protocol we secure the channel using Secure Sockets Layer (SSL) (i.e. HTTPS). This ensures the integrity and confidentiality of assets in LO. To achieve authentication and authorization we rely on the verification of client certificates provided by SSL. In practice, to implement this approach we just needed to configure the servlet container (e.g. Tomcat) to support HTTPS requests with authorized certificates. Nevertheless, managing certificates is a comparatively complex procedure thus we provide a set of auxiliary functions in the core that act as a mini Certificate Authority (CA). These functions are used for managing and signing client certificates and their implementation is based on the Java Security APIs.

6 Tests and Evaluation

Reliability is one of our main concerns regarding the Core component of crimsonHex. We adopted JUnit as our automated unit testing framework since crimsonHex is implemented in Java and this tool is support by Eclipse, the Integrated Development Environment (IDE) used in this project. Apart from the unit tests, we created a tool for automatic generation of random requests to the repository, following the communication model summarized in Figure 3. The goal of this tool is two folded: to look for bugs in unpredicted sequences of requests and to stress-test the repository. The tool generates a random sequence of Core functions' invocations and records then in the Core's log file (through a Java-based logging utility called log4j). Errors generated by these request sequences are recorded by the Core in the same log files. After each test the log file is manually inspected looking for function sequences that

originated errors. This approach was essential to discover errors that otherwise would only be detected in production. Efficiency and scalability are two other main concerns in the development of crimsonHex. To test performance we used the test tool to compare execution times of the main functions in the two supported web services interfaces: SOAP and REST. Each function has been repeated 10 times. Average function execution times for the set of functions are shown in Table 2.

Table 2. Average function execution times per interface (in seconds)

	submit	retrieve	Search
SOAP	4,53	1,57	2,23
REST	2,11	0,44	0,93

These figures show that our DRI extension, based on REST, twice as efficient as the standard SOAP interface. These results were expectable since the REST interface does not have to marshal request messages. In both interfaces submit times are significantly higher than the other functions due to weight of the validation process.

The scalability his other important issue. Scalability is bound by the database limits. The eXist NXD supports a maximum of 2^{31} documents and theoretically, documents can be arbitrary large depending on file system limits, e.g. the max size of a file in the file system, which have an influence. To test the scalability of eXist some queries were made [9] with increasing data volumes. The experiment shows linear scalability of eXist's indexing, storage and querying architecture.

7 Conclusions

In this paper we described the architecture, design and implementation of a repository of specialized learning objects called crimsonHex. The main contribution of this work is the extension of the existing specifications based on the IMS standard to the particular requirements of a specialized domain, such as, the automatic evaluation of programming problems. We focused mainly on two parts:

- the specialization of the definition of LO, where programming problems are given as a concrete example;
- the design of the repository, more precisely, its components, functions and details of its implementation.

For the first part we detail the actions needed to define LOs from a domain that is not covered by the IEEE LOM in a way that can be reproduced in similar contexts.

For the second part we describe the design and implementation of a repository of specialized LOs. We adopt the IMS DRI and propose extensions to its recommendations, namely on the web service interfaces and on the standard functions. The new function to record usage reports of a LO, will be the basis to support a next generation of LMS with the ability to tailor the presentation order of programming exercises to the needs of a particular learner.

In its current status crimsonHex Core can be deployed to a service oriented eLearning platform and is available for test and download at the following URL

http://mooshak.dcc.fc.up.pt:8080/crimsonHex/releases.jsp. Our future work in this project includes developing a management and authoring tool; populating the repository with problem sets from existing sources, while classifying then and controlling their quality.

Acknowledgements. This work is part of the project entitled "Integrating Online Judge into effective e-learning", with project number 135221-LLP-1-2007-1-ES-KA3-KA3MP. This project has been funded with support from the European Commission. This communication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

References

1. Dagger, D., O'Connor, A., Lawless, S., Walsh, E., Wade, V.: Service Oriented eLearning Platforms: From Monolithic Systems to Flexible Services (2007)
2. Girardi, R.: Framework para coordenação e mediação de Web Services modelados como Learning Objects para ambientes de aprendizado na Web (2004)
3. Wilson, S., Blinco, K., Rehak, D.: An e-Learning Framework. Paper prepared on behalf of DEST (Australia). In: JISC-CETIS (UK), Canada (2004)
4. Holden, C.: What We Mean When We Say "Repositories" User Expectations of Repository Systems. In: Academic ADL Co-Lab (2004)
5. JORUM team: E-Learning Repository Systems Research Watch. Technical report (2006)
6. Hatala, M., Richards, G., Eap, T., Willms, J.: The EduSource Communication Language: Implementing Open Network for Learning Repositories and Services. In: ACM symposium on Applied computing (2004)
7. Fielding, R.: Architectural Styles and the Design of Network-based Software Architectures-Phd dissertation (2000)
8. Friesen, N.: Semantic and Syntactic Interoperability for Learning Object Metadata. In: Hillman, D. (ed.) Metadata in Practice, Chicago, ALA Editions (2004)
9. Meier, W.: eXist: An Open Source Native XML Database. In: NODe 2002 Web and Database-Related Workshops (2002)

A Scalable Parametric-RBAC Architecture for the Propagation of a Multi-modality, Multi-resource Informatics System

Remo Mueller, Van Anh Tran, and Guo-Qiang Zhang*

Case Western Reserve University, Cleveland OH 44106, USA
{remo.mueller,vananh.tran,gq}@case.edu

Abstract. We present a scalable architecture called X-MIMI for the propagation of MIMI (Multi-modality, Multi-resource, Informatics Infrastructure System) to the biomedical research community. MIMI is a web-based system for managing the latest instruments and resources used by clinical and translational investigators. To deploy MIMI broadly, X-MIMI utilizes a parametric Role-Based Access Control model to decentralize the management of user-role assignment, facilitating the deployment and system administration in a flexible manner that minimizes operational overhead. We use Formal Concept Analysis to specify the semantics of roles according to their permissions, resulting in a lattice hierarchy that dictates the cascades of RBAC authority. Additional components of the architecture are based on the Model-View-Controller pattern, implemented in Ruby-on-Rails. The X-MIMI architecture provides a uniform setup interface for centers and facilities, as well as a set of seamlessly integrated scientific and administrative functionalities in a Web 2.0 environment.

Keywords: Role-based access control, Scalable information system, Web 2.0.

1 Introduction

A significant challenge encountered by biomedical research facilities today is the efficient management of costly instrumentation, staff time, as well as the storage and archiving of large volumes of complex experimental data and results. According to [2], this challenge is both widespread and acute, and can only become more magnified as facilities grow and new tools and techniques become available, a trend recognized by the NIH Roadmap [8].

There are no off-the-shelf software packages which combat these challenges adequately. To address this research informatics infrastructure issue, we have developed a comprehensive web-based information management system called MIMI (Multi-Modality, Multi-Resource Informatics Infrastructure) that seamlessly integrates administrative support and scientific support in a single system. Separate instances of MIMI have been deployed over the past two years, in three different kinds of facilities: Imaging [14], Proteomics, and Flow-Cytomety. With the deployment of more instances of

* Corresponding author.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 114–124, 2009.
© Springer-Verlag Berlin Heidelberg 2009

MIMI, we realize that scalability and usability are essential properties of a web application in the biomedical research infrastructure domain. Here scalability refers to a system architecture's ability in setting up and managing a large set of administrative workflows and roles in a secure manner in a complex organization without direct, centralized control. Usability refers to strengthened user interface design in order to account for large discrepancies in computer experience among the users of such a system. Web 2.0 allows us to address the usability issue in a way such that a web application such as MIMI compromises neither features nor responsiveness compared to a typical stand-alone desktop application.

This paper presents a scalable architecture called X-MIMI, to provide a streamlined process to propagate MIMI instances to centers and facilities. A key component of X-MIMI is a setup framework that allows a facility to both setup and manage its available services, users, and supporting staff, under the organizational structure of a center. Centers such as NCI designated Comprehensive Cancer Centers [7] play a critical role in translational research, with both pre-clinical and clinical facilities under a common administrative infrastructure to facilitate the translation of discoveries from "Bench to Bedside" [3]. A facility in the X-MIMI architecture is composed of five main entities: people, equipment/service/workflow, input sample, output/raw data, and administration. These entities can be further categorized into different subtypes if necessary.

Access to the setup framework and other content areas of X-MIMI is mediated by a parametric RBAC (Role-Based Access Control [9]) model, PRBAC. Our extension combines both ARBAC - Administrative RBAC [10] and User-to-User Delegation [13] to provide maximal administrative autonomy and flexibility without compromising information security. ARBAC allows the management of RBAC itself as an explicit permission. User-to-user delegation allows for some of the permissions of a user to be routinely exercised by other users, such as sharing data, generating reports, and managing the scheduling of resources. We also employ user-to-user delegation extensively in user interface testing. In order to ease the complexity of applying the PRBAC model in X-MIMI, we use Formal Concept Analysis [5] as a novel semantic framework for PRBAC, allowing the role-hierarchy to be derived from the Role-Permission table automatically to minimize potential inconsistencies between the role-hierarchy and the intended authority of a role.

Another feature of X-MIMI system is using WYDIWYS (What You Do Is What You See [4]) as the guideline in designing an effective user interface. This feature, together with parametric ARBAC, requires a thorough analysis of user privileges and system functionalities to ensure that navigational links and actions are not displayed when users are not authorized to perform. If it is not done systematically, for data and feature rich systems such as X-MIMI, it may result in a poorly organized system that is hard to extend and maintain.

X-MIMI has the following set of features:

- it provides an integrated solution for managing all informatics aspects of centers and their facilities in a single system, from resource scheduling, operations management, user management, project and data management to billing and report generation;
- it is scalable for deployment by using a decentralized setup framework supported by a parametric RBAC model;

- it is user friendly with an WYDIWYS Web 2.0 user interface with rich menu-driven drag-and-drop features;
- it has been developed closely with the end-user in the loop using an Agile development methodology (the discussion of which is beyond the scope of the paper), and has been fully tested and deployed for over two years.

The implementation of MIMI takes advantage of the MVC software design pattern in a Ruby-on-Rails developmental environment [1], which embodies many of the Web 2.0 characteristics. In particular, AJAX and RESTful techniques have been incorporated whenever applicable and appropriate to improve usability of MIMI.

The rest of the paper is organized as follows. Section 2 overviews the organizational structure of a center and conceptualize important roles that must be captured in an informatics infrastructure. Section 3 presents the X-MIMI architecture and discusses its scalability properties and use cases. PRBAC is highlighted as a system level information-flow controller. Section 4 concludes the paper with discussions and future work.

2 Organizational Structure of a Center

We provide an overview of the typical organizational structure of a center. This view is necessarily coarse in order to account for a variety of centers. It provides a basis for our X-MIMI architecture in the next section.

The overall design philosophy of X-MIMI is a system that manages resources to be used by researchers in a structured way. Both the "resources" and the "users" are to be interpreted in the broadest sense. For example, one type of resource may be costly instruments and imaging systems. Another type of resource may be valuable patient and experimental data. Domain-expertise is yet another type of resource that is usually achieved through consultation services. User types may include investigators, business administrators, center directors, instrument operators, and system administrators.

In the biomedical setting, these "resources" and "users" are often organized in the administrative structure of a center (see Figure 1), which consists of one or more facilities. Each facility is an administratively independent unit. The five key aspects of a facility are: people, research resource, sample, scientific data and administration.

The human resources (i.e. people) of a facility can be classified as follows.

Center Administrator. A Center Administrator is responsible for managing user information, project information, center-level reports, and center-level setup. The Center Administrator is in charge of setting up the center cores, institutions, departments, programs and focus groups in the system.

Facility Administrator. A Facility Administrator is responsible for setting up and maintaining a facility's services, equipment, workflows, samples, and invoices. The Facility Administrator can also generate facility-level usage reports.

Principle Investigator (or PI). A PI is a person in charge of a project. A project usually has a unique PI, but several users can assume the PI-role on a project if the original PI designates the PI role to others. The PI has the administrative authority and responsibility for project management, including financial spending and data access. Additional users can participate in project activities with the PI's permission.

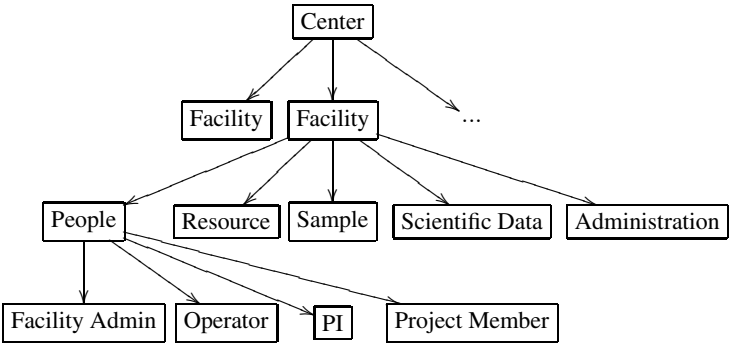


Fig. 1. Organizational Structure of a Center

Operator. An Operator is in charge of completing a service and operating an instrument. Operator also helps a PI and project members to conduct research experiments.

Together, these categories of users make up the roles in our parametric ARBAC.

3 The X-MIMI Architecture

What distinguishes X-MIMI from other existing systems is its seamless integration of administrative support and scientific support in a single system, to be deployed in the complex organizational structure of a research center. With respect to administrative support, X-MIMI's functionalities include the scheduling of instrument systems and resources, tracking resource usage and staff time, billing and accounting, as well as various kinds of usage report generation. With respect to scientific support, X-MIMI's functionalities include project management, workflow management (sequence of experimental steps performed on different instruments) and data management (data archiving, access, and sharing).

Figure 2 is a high-level view of the system architecture for X-MIMI. The next section provides details on each individual component of the X-MIMI architecture.

3.1 X-MIMI System Components

Model-View-Controller. The Model-View-Controller pattern logically separates portions of the core components of a web server [12]. The model serves as the mapping between the database and the object tables.

Web content is organized using the RESTful paradigm. The RESTful paradigm allows logical web site design, and allows for a coherent model for individual Rails controllers. RESTful web page design makes use of the four HTTP verbs: GET, POST, PUT, and DELETE. When combined with structured urls, these verbs create a powerful way of organizing the structure of the controller code, along with making logical access to the model using the Rails functions: *index*, *new*, *create*, *edit*, *update*, and *destroy*. X-MIMI further adds to the RESTful design by integrating it with PRBAC, which allows access control based on roles and permissions along with implicit database relationships that state whether or not a user has the right to access certain resources.

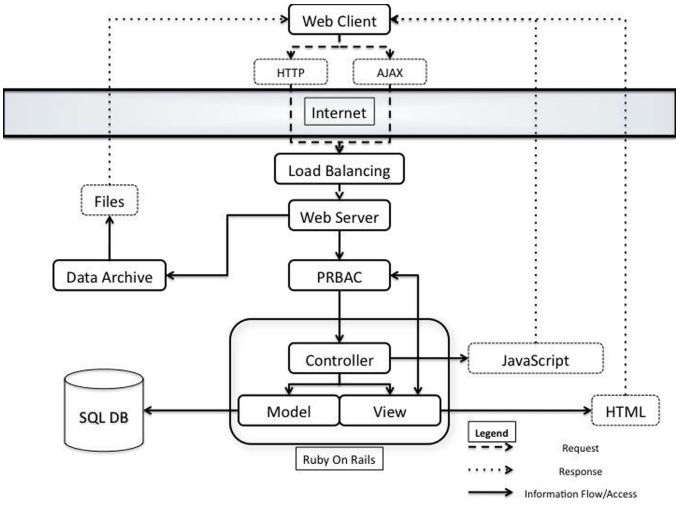


Fig. 2. X-MIMI Architecture

Database. We use ontological modeling to design a database schema that is generic enough to be expanded to many different research centers with minimal modifications.

Figure 3 indicates the main data tables used for MIMI system and the relationships between them implemented to assure the scalability of the X-MIMI system and make it more adaptable. Arrow headed connectors indicate a hierarchical or "part-of" relationship. For example, the arrow from *project* to *study* shows that "a project has many studies", or "a study is a part of a project". Circle headed connectors indicate an attribute relationship. For example, "PI is an attribute of a project" and "grant (or funding-status) is also an attribute of a project".

Different shapes in Fig. 3 indicate different object categories. Oval corresponds to concrete objects; a core is made up of facilities which in turn provide services through the use of equipment. The squared objects are for project organization. Octagons are used for PI and grant, which are somewhat independent of the other objects. The "component" object is shown with a dashed outline in order to indicate that it is invisible anywhere in the user-interface. It is used to group sessions scheduled within a particular facility (i.e. a realization of an experimental workflow which may involve several services in sequence). Component and session are shown with rounded corners to indicate that they are associated with a particular facility (oval) (but project and study are not) though still part of the project hierarchy (square).

Roles and permissions are also stored as entries in the database which allows for new roles to be easily created and added later to the X-MIMI system.

3.2 Data/Information Flow

X-MIMI operates in two modes: Setup Mode and Deployed Mode.

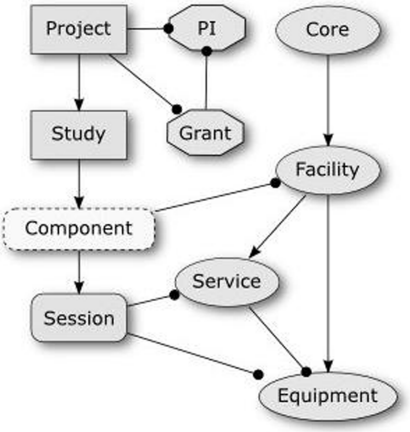


Fig. 3. Relationships among some key terms

Setup Mode. Setup Mode describes the state of the X-MIMI application as administrators create a virtual representation of their cancer center and its facilities. X-MIMI is initialized with a single system administrator active inside the system. The system administrator has the permission to assign a user as a center administrator. The center administrator in turn sets up the cores and facilities, programs, institutions, departments, and focus groups within the system along with assigning users as facility administrators. Each facility administrator can then set up facility specific items such as services, equipment, workflows, and discounts and give users roles as principal investigators or operators within the facility. As soon as services are assigned operators, the system is considered to be in deployed mode. The hierarchy of roles can be seen in Figure 4. X-MIMI can also be set up by migrating information directly from previous MIMIs, such as the Imaging, Proteomics, and Flow-Cytometry MIMIs, or by preloading users from an existing user database, as in the case of loading users from an existing cancer center member database.

Deployed Mode. X-MIMI enters Deployed Mode when the predominant activity consists of users scheduling and requesting time for services. A service that has been scheduled by a user is called a session. A session goes through the following states: pending, approved, completed, invoiced, and audited. A user requests a session (pending) for a particular service. A facility administrator approves and schedules the session (approved). An operator runs the service and completes the session (completed). A facility administrator creates an invoice for the session (invoiced) and bills the principal investigator. An auditor audits the session (audited) at which point the session can no longer be modified.

Sessions form the basis for usage report for equipment, facility and center.

3.3 Parametric Administrative RBAC

Role-Based Access Control (RBAC [9,11]) is a security policy framework for an organizational information system. Permissions (for operations) are associated with roles, and

roles are assigned to users. RBAC provides the ability to assign roles to users dynamically which helps reducing administrative complexity and potential for access errors. To further reduce administrative complexity and aid scalability, we use a combination of Administrative RBAC ARBAC [10] and User-to-User delegation system [13].

In ARBAC, the administration of roles is included as a permission, so that roles higher up in the security hierarchy can assign roles to users that are lower in the hierarchy. In X-MIMI, this feature will allow the center administrator to delegate much of the administrative tasks to facility administrators, including the assignment of operators for instruments, admitting PIs to use resources in the facility. We further extend ARBAC to a parametric ARBAC to improve the scalability of X-MIMI. A role is described in the form of $Role(i_1, ..., i_n)$ in which i_j represents a facility, a service or an equipment... For example, Facility Administrator(flow cytometry) is assigned to users who have Administrator role in Flow Cytometry Facility. User who has Facility Administrator in facility i may have only role of baseline user in other facilities. Another example, Operator(s,i) is operator on instrument s in facility i.

User-to-User delegation allows a grantor to delegate part or all of his roles and resources to another user. A delegation has the form $delegation(U1, U2, R, n)$ where $U1$ is the grantor and $U2$ is the proxy. R represents the roles and resources given to the proxy. The number n indicates that the proxy will have the ability to further delegate R to another user who can further delegate in no more than $n-1$ steps. When $n = 1$, only immediate delegation is permitted. In X-MIMI, this delegation is limited to one step, so that the proxy (the user who is being delegated) does not have the right to further delegate the delegated roles and resources to another user.

The application of ARBAC requires a predefined role-hierarchy. Ensuring the consistency of this hierarchy and role-permission table, so that a role in the upper hierarchy does not have less permissions than a role in the lower hierarchy, is a topic that has not been adequately addressed systematically in the past. We use Formal Concept Analysis (FCA [5]) as the mathematical framework to address this issue. FCA provides a general framework for translating a binary relation such as a role-permission table called "context" into a lattice, with nodes representing closed-sets or concepts, and links among them representing subsumption. In PRBAC, such a lattice determines the desired role-hierarchy which is guaranteed to be consistent with the starting role-permission table, as a consequence of the general property of FCA. The details on how this works are beyond the scope of the current paper, though we plan to publish the results elsewhere.

For example, in our implementation (see Section 4), we use the following as part of the role-permission table for RBAC.

This systematically generates a role-hierarchy using any of the software tools for FCA (such as ConExp). The cascade of role-assignment permission matches perfectly with our application needs in the center-facility setting, which the ARBAC permissions consistent with the role-permission table.

Developmental. The proxy system allows a continuation of rapid development even with many different roles. A developer can easily switch between users that have different roles without the need for logging in and logging out as different users. This also helps a developer to track down bugs reported by an individual user by being able to replicate the exact steps while logged in as the target user.

Table 1. Role-Permission Table for X-MIMI

	Global	Manage Project	Complete Session	Administrate Session	Setup Center	Setup Facility
Center Administrator	×	×			×	
Facility Administrator		×		×		×
Operator			×	×		
Principle Investigator		×				

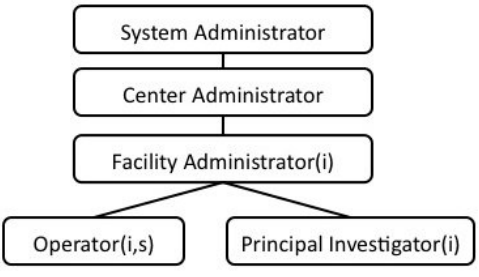


Fig. 4. Role Hierarchy in MIMI

End User. Adding a proxy allows a user to set up another user to imitate them. An example is a principal investigator who would like one of his graduate students to occassionally perform the principal investigator tasks in his place. The principal investigator may not have the time to do these tasks, or the principal investigator might not be knowledgeable enough about the system to do these tasks. Therefore delegating the tasks to a different user is useful. The advantage of having a proxy is that the user does not need to give the proxy his password. The proxy feature also allows for distributed input from different people which would allow the database information to stay up-to-date.

3.4 WYDIWYS Web Inferface

To improve user experience for users in research centers we implemented a WYDIWYS web interface for X-MIMI [4]. Each user will have different interfaces depending on their roles. We first break functionalities down to basic actions corresponding to data accesses. The MVC design pattern offered by Ruby-on-Rails allows us to define four basic actions for each data entity: *index*, *create*, *update*, and *destroy*. Similar actions are grouped into privileges.

Figure 5 shows part of the scheduling interface that is available to a Principal Investigator. By default, the scheduling interface shows scheduled sessions accessible by the Principal Investigator as well as unavailable times. A Facility Administrator sees all sessions scheduled in the facility. The property of WYDIWYS is achieved through the use of partials which reduces the number of views that would otherwise need to be created.

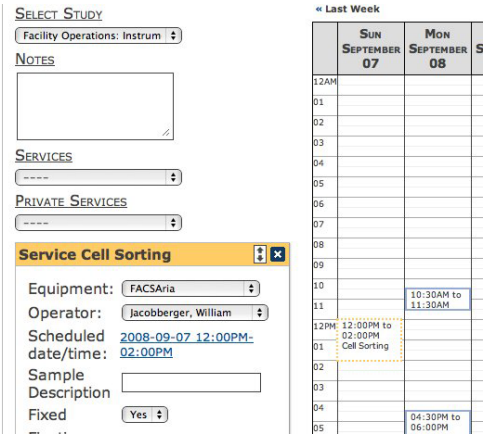


Fig. 5. Principal Investigator Scheduling a Session

3.5 Role-Based Testing

Ruby-on-Rails allows the user to create unit, functional and integration test cases to test certain features. These test cases can be performed automatically and allow the developer to easily discover regression bugs caused by refactoring and optimizing code.

Example test cases for MIMI include user login and registration, user access to restricted portions of the website based on the user's role, and testing role assignment from a higher level user to another.

A role is defined by the set of privileges that is assigned to the role. Unit tests are therefore performed on the individual privileges themselves. These unit tests cover control access, to determine that each low level function can only be performed by the appropriate privilege. For instance, only a user with the *Manage Project* privilege is allowed to edit project details and assign project users.

Functional role-based testing more complex steps that involve privileges. Functional testing includes role assignment from one user to another. In this case, the functional tests assure that users cannot assign roles that have more privileges that the original user.

Finally integration tests verify that individual webpages display the correct information for users with multiple roles, and therefore multiple privileges. Integration tests need to take into account many more factors than just roles and privileges, such as in which facility the user is currently active.

Figure 6 shows a Proteomics facility administrator assigning the Principal Investigator role to a user. The administrator does not have the privilege to assign roles in other facilities, or to assign roles that contain higher privileges than the administrators privilege.

3.6 Experimental Results

To demonstrate the feasibility of the X-MIMI system, X-MIMI has been being deployed in one of 39 NIH-designated cancer centers which provides a nexus for coordinated

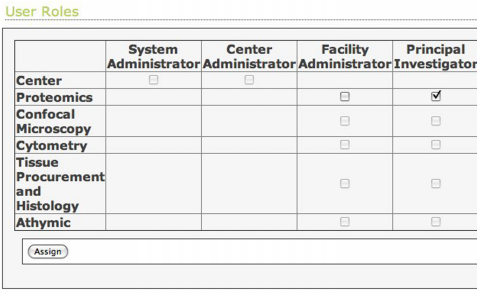


Fig. 6. Facility Admin Assigning New Roles

interdisciplinary research into all aspects of cancer by facilitating interactions among laboratory, clinical, translational and population scientists. The Center supports 17 shared resources that facilitate cancer related research conducted by 350 faculties of two universities and two well-known medical facilities in 9 scientific programs. One of the essential characteristics of an NCI-designated Cancer Center is shared resources, managed separately with different policies through *core facilities* [6].

Two previous instances of MIMI have also been deployed: one in the Imaging core facility, and the other in the Proteomics core facility. Since its deployment in May 2006, the Imaging-MIMI has 185 registered users working on 148 different projects, many of them are sponsored by external funding. About 2TB of imaging data have been archived in the data-server, resulting from a total of 2827 sessions. The Proteomics-MIMI was deployed in July 2007. In about just 8 months, 109 users on 97 distinct projects have been managed through the Proteomics-MIMI with 292 studies consisting of 553 total sessions. This shows that MIMI can be used to process a myriad of data as well as a large number of users, processes. MIMI 2.0 has been being implemented with four main roles: *Center Administrator*, *Facility Administrator*, *Principle Investigator* and *Operator*. Other roles can be easily added with different permissions when needed.

4 Conclusions

We find that parameterized RBAC is a highly useful tool for developing complex web-based, informatics systems. The architecture of X-MIMI and the extensive use of role-based software testing helped us to create a robust system for managing the latest instruments and resources used by clinical and translational investigators. WYDIWYS using FCA allowed for intuitive and minimal user interface design. Ruby-on-Rails allowed us to create a fully functional system with impressive developmental productivity.

Due in part to this initial success of MIMI, the NIH recently funded a multi-institution project called Physio-MIMI to our university [15]. The X-MIMI architecture is also featured in a Request Management System under development for the CTSC [16].

Acknowledgements. Funding support for this project is provided by the Case Comprehensive Cancer Center. Thanks go to James Jacobberger, Anne Duli and William Jacobberger for their contribution to system requirement specification. Additional members

of the MIMI developer team include Jacek Szymanski and Jie Dai. The project described was also supported in part by Grant K25EB004467 from NIBIB. Its contents are solely the responsibility of the authors and do not necessarily represent the official views of the NIBIB or NIH.

References

1. Ruby on Rails, http://www.rubyonrails.org
2. Anderson, N., Lee, E., Brockenbrough, J., Minie, M., Fuller, S., Brinkley, J., Tarczy-Hornoch, P.: Issues in biomedical research data management and analysis: Needs and barriers. J. Am. Med. Inform. Assoc. 14, 478–488 (2007)
3. Clinical and Translational Science Awards, http://www.ncrr.nih.gov/clinical_research_resources/clinical_and_translational_science_awards
4. Dai, J., Mueller, R., Szymanski, J., Zhang, G.-Q.: Towards "WYDIWYS" for mimi using concept analysis. In: The 24th Annual ACM Symposium on Applied Computing (in Press, 2009)
5. Ganter, B., Wille, R.: Formal Concept Analysis (1999)
6. National Cancer Institute - The Cancer Centers branch of the National Cancer Institute: Policies and guidelines relating to the Cancer Center support grant, http://cancercenters.cancer.gov
7. NCI Designated Cancer Centers, http://cancercenters.cancer.gov/cancer_centers/cancer-centers-list.html
8. NIH Roadmap for Medical Research, http://nihroadmap.nih.gov
9. Park, J., Costello, K., Neven, T., Diosomito, J.A.: A Composite RBAC Approach for Large, Complex Organizations. In: SACMAT 2004 (2004)
10. Sandhu, R., Bhamidipati, V., Munawer, Q.: The ARBAC97 Model for Role-Based Administration of Roles. ACM Trans. Inf. Syst. Secur. 2(1), 105–135 (1999)
11. Sandhu, R., Coyne, E., Feinstein, H., Youman, C.: Role-Based Access Control Models. IEEE Computer 29(2), 38–47 (1996)
12. Sauter, P., Vögler, G., Specht, G., Flor, T.: A Model–View–Controller Extension for Pervasive Multi-Client User Interfaces. Personal Ubiquitous Comput. 9(2), 100–107 (2005)
13. Wainer, J., Kumar, A.: A Fine-Grained, Controllable, User-to-User Delegation Method in RBAC. In: SACMAT 2005, vol. 9(2), pp. 59–66 (2005)
14. Szymanski, J., Wilson, D.L., Zhang, G.-Q.: MIMI: Multimodality, Multiresource, Information Integration Environment for Biomedical Core Facilities. Journal of Digital Imaging (in press) (2007), doi:10.1007/s10278-007-9083-y
15. Three New Informatics Pilot Projects to Aid Clinical and Translational Scientists Nationwide, http://www.nih.gov/news/health/jan2009/ncrr-26.htm
16. Clinical and Translational Science Collaborative, http://casemed.case.edu/ctsc

Minable Data Warehouse

David Morgan[1], Jai W. Kang[1], and James M. Kang[2]

[1] College of Computing and Information Sciences, Rochester Institute of Technology
Rochester, NY, USA
davidmorgan@mail.rit.edu, jai.kang@rit.edu
[2] Department of Computer Science, University of Minnesota, Minneapolis, MN, USA
jkang@cs.umn.edu

Abstract. Data warehouses have been widely used in various capacities such as large corporations or public institutions. These systems contain large and rich datasets that are often used by several data mining techniques to discover interesting patterns. However, before data mining techniques can be applied to data warehouses, arduous and convoluted preprocessing techniques must be completed. Thus, we propose a minable data warehouse that integrates the preprocessing stage in a data mining technique within the cleansing and transformation process in a data warehouse. This framework will allow data mining techniques to be computed without any additional preprocessing steps. We present our proposed framework using a synthetically generated dataset and a classical data mining technique called Apriori to discover association rules within instant messaging datasets.

Keywords: Data warehouse, Data mart, Data mining, Apriori, Association rule mining.

1 Introduction

Motivation. Many corporations all over the world maintain their valuable historical datasets using data warehouses that collect an abundant amount of information. Due to the integrated and cleansed information stored in data warehouses, these systems have been one of the major sources for data mining techniques. A popular data mining technique that often uses these types of datasets is association rules. Association rule mining (ARM) identifies sets of object types that may co-occur together more often than other item types. Thus, it is crucial that the information stored in a data mart must be readily accessible to these data mining methods to determine specific patterns quickly and efficiently.

Problem Description. Given one or more data marts in a data warehouse, the goal is to have data readily available for a data mining algorithm. The main objective is to reduce the amount of effort to prepare the dataset for data mining methods. For example, in ARM of market basket datasets, a crucial criterion of the input to ARM is to have transactional information containing a unique transaction id and the various item types for each transaction.

Challenges. Preparing the data from a data mart quickly for a data mining technique is extremely challenging for several reasons. First, information stored in a data mart for

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 125–136, 2009.
© Springer-Verlag Berlin Heidelberg 2009

a certain business unit is often prepared based on their own needs and is not suitable for direct use of data mining techniques (e.g. association rule mining). Second, the preprocessing time for data mining methods is often convoluted and may remove pertinent information that may be crucial for the resulting pattern set. Finally, as the datasets in a data mart get updated, the preprocessing step needs to be re-computed before the data mining technique can execute.

Related Work. Several researchers and analysts have claimed that a data warehouse may be an excellent data source for data mining methods and have postulated that these two can create a "symbiotic relationship" [5,8]. They have not explored how a data warehouse may be used for data mining specifically. For example, ARM has been used in data marts to refine the set of patterns but has not addressed the preprocessing problem that occurs between each system [7]. Likewise, materialized views in data warehouses have been used for data mining methods to allow for repetition of methods [2] but have not examined the preprocessing steps to produce the end result. Modifying data warehouses to accomodate data mining methods has been explored by maintaining a single attribute using bit mapped indexes [10] but has not been shown directly to handle multiple attributes for association rule mining. Thus, to the best of the authors' knowledge, no previous works have explored the integration of data mining and data warehouse by reducing the amount of preprocessing for multi-attributes.

Contributions. In this paper, we propose a novel framework that allows data mining techniques to access a data mart without any preprocessing steps in between. In general, data that are stored in a data mart must go through an entire cleansing and transformation process to prepare the data before they are stored in the system. Likewise, data mining techniques must also perform some form of cleansing or preprocessing before the technique can be executed. Thus, we propose to re-construct the traditional cleansing and transformation process in data warehouses to align with the requirements in data mining techniques. This would allow an efficient streamline between data warehouses and data mining techniques for quick and efficient pattern sets. We evaluate this framework using a synthetic dataset of instant messaging and the classical association rule mining algorithm called Apriori. In summary, this paper makes the following contributions:

1. We propose a novel framework called a *Minable Data Mart* that will allow data mining techniques to perform on a data mart without the use of any preprocessing steps.
2. We evaluate our framework using a case study that utilizes the Apriori method to discover associations in a transactional dataset.
3. We present our system with several screen shots to illustrate the proposed framework.

Organization. The rest of the paper is organized as follows. Section 2 presents the general basic concepts and an example used throughout this paper. Three different frameworks are given, specifically *Data Mining without Data Warehouses*, *Data Mining with Data Warehouses*, and *Mineable Data Warehouse* in Sections 3, 4, and 5 respectively. A demonstration of the mineable data warehouse is given in Section 6. Finally, Section 7 concludes the paper and discusses future work.

2 Basic Concepts

This section presents the basic concepts that are used throughout this paper. First, we introduce several basic concepts in data warehouses. Second, we highlight one of the classical data mining approaches we use as part of our examples and within the demonstration. Finally, we give an example using a synthetic dataset of instant messaging with data warehouses and data mining.

2.1 Data Warehouses

In general, a data warehouse is an integrated system that is designed to facilitate user analysis. This system provides a single, clean, and consistent source of historical data for many types of decision making. Also, it provides strategic information based on an enterprise-wide scale.

Data warehouses can be developed as either a top-down [5] or a bottom-up [9] approach. In a top-down approach, a data warehouse is initially defined as the entire organization where each faction may have a different operation (i.e., Enterprise Data Warehouse). Data Marts are extracted from the top-down creation of the data warehouse. However, in a bottom-up approach, one business unit may create their own data mart (e.g. sales, marketing, etc.) first. Then, the next business unit creates their own data mart which conforms to the dimensions in the previously created data mart. This process continuous for all business units in a corporation until the final data warehouse is completed.

Data Warehousing has traditionally been used for on-line analytical processing (OL AP) systems [6,8]. Data that are stored in a data warehouse or data mart must go through an entire cleansing and transformation process to be stored and ensure that the data is clean [5,9]. Data miners must also go through a similar cleansing and preparing process to run their mining algorithms on the same source data. It has been said that a data warehouse would provide an excellent base for data miners to use as source data because of its cleaned nature [9].

A data mart is a dimensional model that represents a single business process in an organization. Its dimensional data must be atomic and should also be built using conformed dimensions in relation to other existing or future data marts. Generally speaking, a data mart is developed using a star schema with a central fact table connected to outlying dimensions [9]. These dimensions will allow for “drilling down” type queries to determine the underlying facts for particular dimensional combinations. Data marts are generally used for OLAP analysis. This analysis helps to inform decision makers about making key business decisions. Multiple conformed dimensions can also be shared with other data marts to build a data warehouse.

An essential step to create a data warehouse is the ETL (Extract, Transform, and Load) phase that has three main steps [5,9]. First, the data is extracted from operational or external data stores. Then, the data is transformed by performing cleansing, aggregation, summarization, integration, and coding transformations. Finally, the data are loaded into the data warehouse. The main goal of this entire process is to obtain clean, consistent, integrated, and possible summarized data.

2.2 Association Rule Mining

One of the classical techniques in data mining is the Apriori algorithm used for association rule learning [1]. In general, association rule learning attempts to discover rules or sets various types that tend to occur together more often than other variables. For example, one of the popular applications for association rule mining is on market basket datasets. In general, market basket datasets may contain itemsets of multiple different types (e.g., various products bought at a grocery). The goal of association rule mining is to discover rules where a certain sets of items are often bought together, i.e., when a certain item type is seen, another item will also be seen. There are an extensive number of algorithms that discover association rules (e.g., see [4,3]), but one of the classical and fundamental techniques is the Apriori algorithm. One of the common problems used for the Apriori algorithm is to determine which sets of items types occur together more often than other combinations of item types. Identifying which items types that will be examined is based on the candidate generation method. Within the Apriori technique, there are two basic interest measures that are often used called *support* and *confidence*.

Support is the general measure that determines how often an itemset occurs within the transaction dataset. For example, suppose we have three transactions having a set of items: $T1(A, B, C), T2(B, C), T3(A, B)$. Out of these three transactions, there are three different item types, A, B, and C. A support threshold may be applied after each size of an itemset is generated, where any itemset that has fewer than the support threshold is not considered the next candidate generation. Suppose the support threshold was 1. First, there are three singletons (i.e. one item type) datasets. Based on the notation of itemset:support, the singletons have the following: $A : 2$, $B : 3$, $C : 2$. All singletons satisfy the support threshold and they can all be used for candidate generation. The next set of candidate itemsets is of size two: $(A, B), (A, C), (B, C)$, where order is irrelevant. The support of each itemset is found as: $(A, B) : 2$, $(A, C) : 1$, $(B, C) : 2$. Since itemset (A, C) does not satisfy our support threshold, this itemset is removed from our answer list. Based on the remaining two itemsets, a final itemset can be created of size three: (A, B, C). This itemset has a support of only one and does not satisfy our threshold. In general, the set of answers is the itemsets that are the longest and satisfies the support threshold. Thus, the final answers in this example are: (A, B) and (B, C).

Confidence is an interest measure used in Apriori that determines whether a certain "rule" best represents the transactional dataset. For example, based on the frequent itemsets we discovered based on the support threshold, (A, B) and (B, C), we can generate a set of possible rules in the form of: $A \Rightarrow B$, $B \Rightarrow A$, $B \Rightarrow C$, and $C \Rightarrow B$. For rule $A \Rightarrow B$, the confidence can be determined by first calculating the number of times item B occurs whenever an item A also occurs by the total number of transactions item A occurs. Thus, the confidence for this rule is: $2/2 = 1$ which means that for every transaction where the item A occurs, item B also occurs. Using the notation of "rules::confidence", the confidence of each rule in our example is: $A \Rightarrow B$::2/2, $B \Rightarrow A$::2/3, $B \Rightarrow C$::2/3, and $C \Rightarrow B$::2/2. Suppose our confidence threshold is 1, then the rules that satisfy are $A \Rightarrow B$ and $C \Rightarrow B$.

2.3 Example

In this paper, we use a synthetically generated dataset that is populated in a data warehouse and then accessed directly by a data mining method. Out of simplicity, the main example used throughout this paper is an instant messaging dataset that was synthetically generated and is mined by using the Apriori approach within within a data mining toolkit called Weka [12]. Other forms of datasets may be applied, and other data mining techniques can be applied to our framework.

Instant messaging datasets contain messages between users and within each message contain a set of words. Identifying associations between words may have several interesting applications such as identifying strong relationships between users, predictive text, etc. Instant messaging datasets can be modeled by having a message as transaction, where each word is an item type. Since this paper simply demonstrates that a data mining technique can use a data warehouse directly without any preprocessing, the actual types of preprocessing such as identifying synonyms and the removal of stop words are not considered in this paper.

3 Data Mining without a Data Warehouse

Several data mining techniques can be performed without the use of a data warehouse. In general, some form of a dataset is provided to the data miner that could by either synthetic or real. One of the main steps before a data mining algorithm can be applied is the preprocessing stage. The preprocessing stage may be different depending on the type of the algorithm used and the actual format needed by the algorithm.

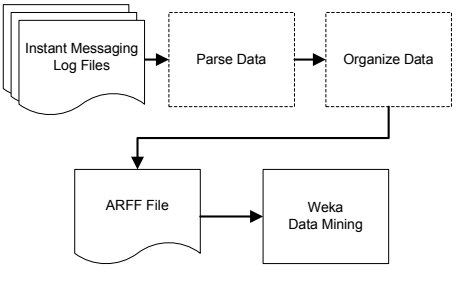


Fig. 1. Data Mining w/o Data Warehouse

Figure 1 depicts an example framework of performing a data mining algorithm without the use of a data warehouse. In this example, the input dataset contains instant messaging logs that may consist of information such as the user name, timestamp of the message, the actual message itself, etc. One of the basic steps in preprocessing is to parse the data to obtain which part of the log files is the user name, timestamp, or the message. Once the data is parsed, further preprocessing is completed to clean (i.e., remove invalid data) the data and re-format the file. In this example, the data is reformatted to an Attribute-Relation File Format (ARFF) [12] that can then be used within the Weka.

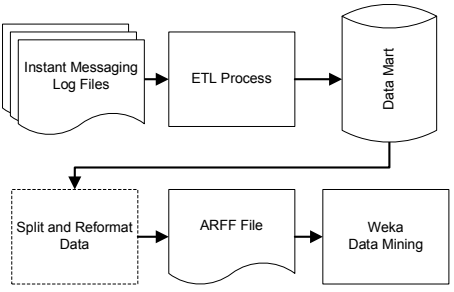


Fig. 2. Data Mining w/ Data Warehouse

4 Data Mining with a Data Warehouse

Data warehouses can be an excellent source for rich and cleaned datasets, and are commonly used for several data mining algorithms. Datasets in a data warehouse can naturally be cleaned using the ETL process. However, data from a data mart may not be able to be directly used for a data mining method. Thus, additional preprocessing may be required to conform the data to the required format for a data mining algorithm.

Figure 2 gives the general framework of performing a data mining method using a data warehouse. Unlike in Figure 1 where there is no data warehouse, Figure 2 does not need a separate step to parse the dataset. Rather, the dataset can be parsed and cleaned as one of the steps in building a data warehouse. Once the datasets are cleaned, additional preprocessing is required to conform the dataset in the required ARFF format that is later used in Weka.

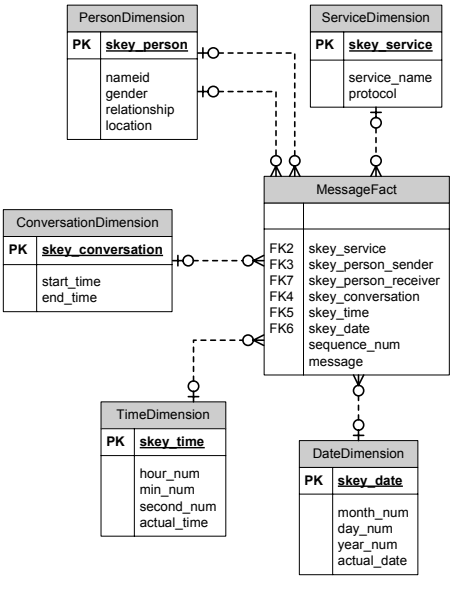


Fig. 3. Initial Mart Design

Figure 3 gives an example of the initial data mart design using the instant messaging dataset. The *grain* of the fact table is a message, and there are several dimensions including the messaging protocol (i.e., ServiceDimension), user name (i.e., PersonDimension), duration of the conversation (i.e., ConversationDimension), time of the message (i.e., TimeDimension), and the date of the message (i.e., DateDimension). It is important to note that one of the attributes within the fact table is "message" which contains the entire message for this user at this time. Further preprocessing will be required by the data miner to split each word in the message and format it for use in Weka.

5 Proposed Framework: Minable Data Warehouse

One of the main limitations in frameworks in Figures 1 and 2 is the aspect of creating additional preprocessing to manipulate the dataset. It is obvious as there are several manipulations done without a data warehouse, but even with a data warehouse, there is still additional preprocessing to parse each word in the messages. Thus, we propose a framework where all unnecessary preprocessing is removed in the entire process and that the data miner can access the data from the data warehouse directly and with ease.

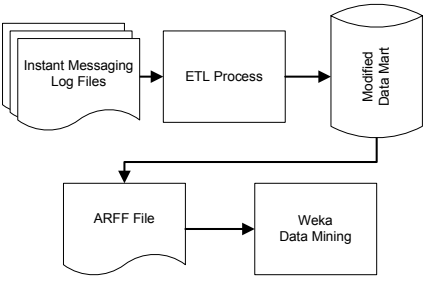


Fig. 4. Mineable Data Warehouse

Figure 4 gives our proposed framework to remove any unnecessary preprocessing between the data miner and the data warehouse. As in the second framework (Figure 2), the proposed framework uses the ETL process to ensure that the data is cleaned. However, by simply reducing the *grain* from the message to the word level will not make the data mart minable due to having a single attribute. For example, the Apriori technique needs a transaction that contains a set of words for each message. When we implement these words as a separate dimension, we have a many-to-many (M:N) relationship between the fact table and the word dimension table. A bridge table may be used to connect multiple words [9,11] to a message. The bridge table allows access to the fact table at a lower grain (word), and thus no additional pre-preocessing is required by the Apriori approach. The only preprocessing that is required is to produce the ARFF file for the Weka system.

Figure 5 gives an example of the logical design of our proposed framework. The key difference between our proposed framework and Figure 3 is the bridge table that links between the fact table and the word dimension. The word dimension contains all the words within each message. The data mart using a bridge table would allow the

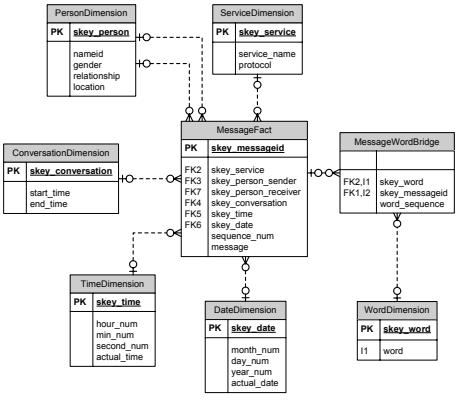


Fig. 5. Data Mart with Bridge Table

creation of an ARFF file without any additional preprocessing to parse the words from the message as in the framework in Figure 2.

5.1 Design Decision

A key design decision is proposed to eliminate all aspects of preprocessing between the data miner and the data warehouse. The general idea is to have all available information within the data warehouse itself and allow for a system such as Weka to directly access it.

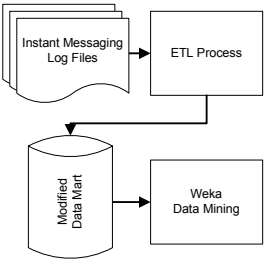


Fig. 6. Design Decision

Figure 6 gives the proposed framework using the design decision. The main difference between this framework and the one in Figure 4 is that an ARFF file does not need to be generated or preprocessed before a data mining system such as Weka can be executed. This is possible due to the structure of the data mart within the logical diagram.

Figure 7 gives the logical diagram of our proposed framework using the design decision. Essentially, the fact table is widened to contain all the words as separate columns in the message. Each word is in a bitmap format to reduce the space needed for each message and for a certain word occurring in the message. This allows a system such as

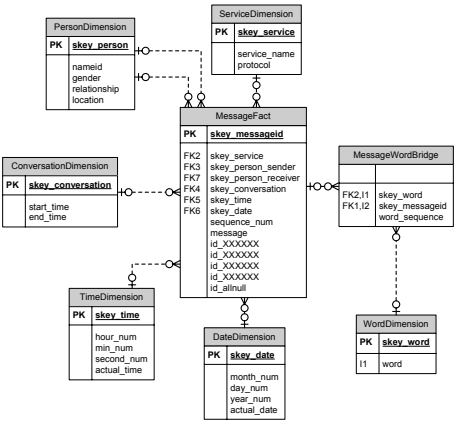


Fig. 7. Final Modified Data Mart

Weka to query the data warehouse directly and extract the data as a single transaction in its standard format (i.e., ARFF). Although this design decision will improve productivity significantly over previous frameworks, maintenance cost may be increased due to the additional columns in the fact table. Further partitioning may be required to reduce the number of words in the fact table and will be explored for future work.

6 Demonstration

In this section, we present a demonstration of our proposed framework (Figure 4) using an instant messaging dataset, Weka, and the output of the association rules. This demonstration illustrates that a data mining technique such as Apriori can directly access a data warehouse and produce association rules.

6.1 Input Files: Instant Messaging Files

Figure 8 gives an example of the synthetically generated instant message files that we used within our proposed framework. The input file contains the user name, the time stamp the message occurred and the message string. Since the proposed work focused on the general framework of using data warehouses and data mining, we simply used each word in the message.

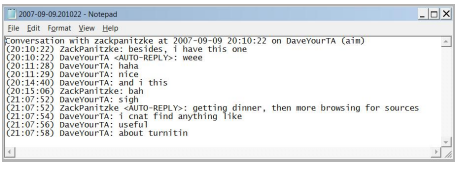


Fig. 8. A Sample Instant Messaging Log File (Best Viewed in Color)

6.2 Input to the Data Mining Tool: Weka

In our proposed framwork (Figure 4), we used the instant messaging files (Figure 8) and loaded them into the data warehouse. Under this proposed framework, we can produce an ARFF file which contains the words in the instant message file in terms of its bitmaps (Figure 9).

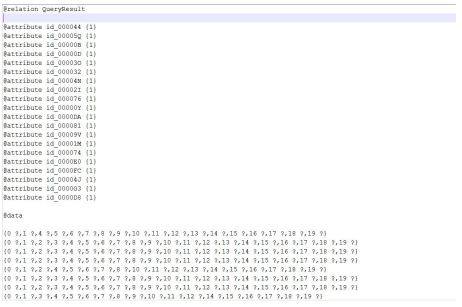


Fig. 9. ARFF File (Best Viewed in Color)

Based on our proposed approach using the design decision, another alternative approach in accessing the information from the data warehouse is by performing a query directly using the Weka system as shown in Figure 10. In this approach, we posed the following query "select * from messagefact", where "messagefact" is the fact table. This query will extract all the messages and its words into the Weka system.

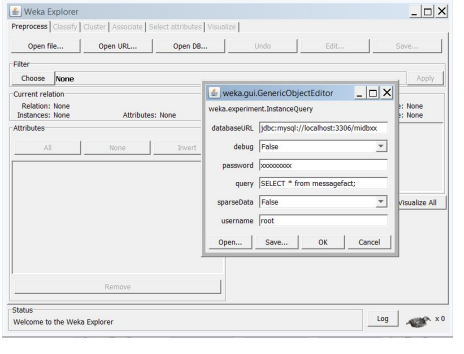


Fig. 10. Access Information from Data Warehouse Directly (Best Viewed in Color)

6.3 Files Viewed on Weka

Based on either using the ARFF file that was generated based on our proposed framework (Figure 4) or using our design decision (Figure 6), the information can then be viewed in Weka (Figure 11). The result illustrates the messages with its respective words on the left pane in the bitmap format along with the count of each word in the right pane in Figure 11.

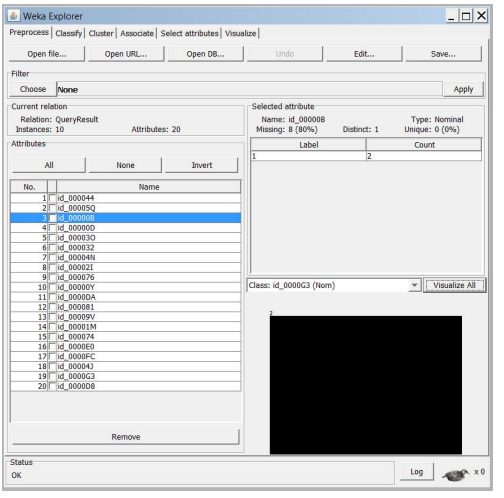


Fig. 11. Files Viewed on Weka (Best Viewed in Color)

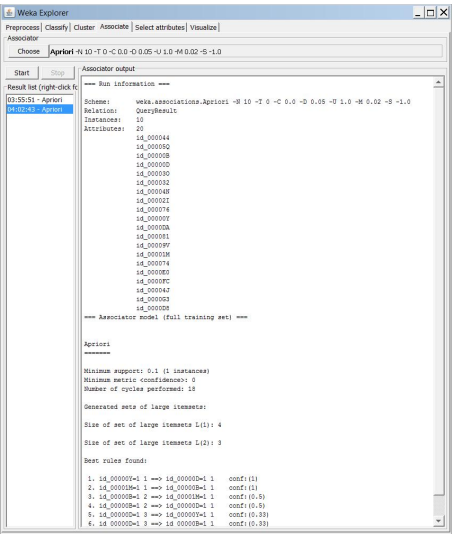


Fig. 12. Generated Association Rules (Best Viewed in Color)

6.4 Generated Association Rules

Figure 12 gives the generated set of rules produced by Weka based on the information provided by our proposed framework. The rules are located at the bottom portion of this figure. It is important to note that the illustration of these rules is simply to show that our framework has the capability to produce data mining patterns directly and is not intended to show the quality of the results. Thus, we can show that Weka can access the data warehouse directly and produce the following association rules in Figure 12.

7 Conclusions and Future Work

In this paper, we proposed a new framework that reduces the amount of preprocessing time that is required by the data miner when using a data warehouse as the main data source. We also present a design decision to remove all aspects of preprocessing to allow for an increased amount of efficiency to obtain data mining patterns. We evaluated our framework using sythetically generated instant messaging datasets and applied it using an Apriori method for association rule mining. We also presented a demonstration of our work using Weka.

We plan to explore alternative methods to improve the maintenance costs of our design decision while maintaining the efficiency to obtain the data mining patterns. Also, we plan on examining other forms of datasets that may be used for data mining techniques. Finally, we plan to generalize our framework to allow for other forms of data mining techniques to be applied by the proposed framework.

References

1. Agrawal, R., Imielinski, T., Swami, A.: Mining association rules between sets of items in large databases. In: ACM SIGMOD International Conference on Management of Data (1993)
2. Czejdo, B., Morzy, M., Wojciechowski, M., Zakrzewicz, M.: Materialized views in data mining. In: 13th International Workshop on Database and Expert Systems Applications, p. 827 (2002)
3. Han, J., Pei, J., Yin, Y.: Mining frequent patterns without candidate generation. In: ACM SIGMOD International Conference on Management of Data (2000)
4. Hipp, J., Guntzer, U., Nakhaeizadeh, G.: Algorithms for association rule mining – a general survey and comparison. In: ACM SIGKDD Explorations Newsletter, vol. 2, pp. 58–64 (2000)
5. Inmon, W.H.: The data warehouse and data mining. Communications of the ACM 39(11), 49–50 (1996)
6. Inmon, W.H.: Building the Data Warehouse. John Wiley & Sons, Chichester (2002)
7. Jukic, N., Nestorov, S.: Comprehensive data warehouse exploration with qualified association-rule mining. Decision Support Systems 42(2), 859–878 (2006)
8. Kimball, R., Reeves, L., Ross, M., Thornthwaite, W.: The Data Warehouse Lifecycle Toolkit: Expert Methods for Designing, Developing, and Deploying Data Warehouses. John Wiley & Sons, Chichester (1998)
9. Kimball, R., Ross, M.: The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling, 2nd edn. John Wiley & Sons, Chichester (2002)
10. Mclaren, I.: Designing the data warehouse for effective data mining (1998)
11. Song, I., Rowen, W., Medske, C., Ewen, E.: An analysis of many-to-many relationships between fact and diemension tables in demensional modeling. In: International Workshop on Design and Management of Data Warehouses (2001)
12. Witten, I.H., Frank, E.: Data Mining: Practical machine learning tools and techniques, 2nd edn. Morgan Kaufmann, London (2005)

A Step Forward in Semi-automatic Metamodel Matching: Algorithms and Tool

José de Sousa Jr[1], Denivaldo Lopes[1], Daniela Barreiro Claro[2], and Zair Abdelouahab[1]

[1] LESERC, Electrical Engineering Department
Federal University of Maranhão, Av. dos Portugueses, s/n São Luís - MA - Brazil
{jgeraldo,dlopes,zair}@dee.ufma.br
http://www.leserc.dee.ufma.br/
[2] Distributed Systems Laboratory (LaSiD), Computer Science Department
Federal University of Bahia, Av. Adhemar de Barros, s/n Salvador - Bahia - Brazil
dclaro@ufba.br
http://www.lasid.ufba.br/

Abstract. In recent years the complexity of producing softwares systems has increased due the continuous evolution of the requirements, the creation of new technologies and integration with legacy systems. When complexity increases the phases of software development, maintenance and evolution become more difficult to deal with, i.e. they became more subject to error-prone factors. Recently, Model Driven Architecture (MDA) has made the management of this complexity possible thanks to models and the transformation of Platform-Independent Model (PIM) in Platform-Specific Models (PSM). However, the manual creation of transformation definitions is a programming activity which is error-prone because it is a manual task. In the MDA context, the solution is to provide semi-automatic creation of a mapping specification that can be used to generate transformation definitions in a specific transformation language. In this paper, we present an algorithm to match metamodels and enhancements in the MT4MDE and SAMT4MDE tool in order to implement this matching algorithm.

Keywords: Metamodel matching, Algorithm, Mapping specification.

1 Introduction

The software production process in MDA is based on models and model transformation. In the MDA context, some research and proposals of transformation languages are available in the literature [1,2,3,4] or in the form of products.

However, the manual creation of transformation definition remains a programming activity that is tedious and error-prone. Thus, an approach that makes the automatic generation of transformation definition possible can leverage the MDA domain.

The task of creating transformation definitions between models is preceded by a mapping specification that consists of searching elements which are semantic and/or syntactic equivalent (or similar) between the target and source metamodel. However, the creation of mapping specification between two metamodels is not an easy task, because metamodels are generally created with different goals and by different development

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 137–148, 2009.
© Springer-Verlag Berlin Heidelberg 2009

teams. This leads to a structural and semantic distance among metamodels that are called *gaps* among metamodels [5].

The manual creation of mapping specification is an activity that becomes more difficult and time consuming as the relationships between metamodels are more complex, i.e., more effort is required to find them.

The creation of mapping specification is also an important task in the fields of database such as database integration, E-business and data warehousing. The correspondence between *schemas* in different databases is called *schema matching* and it plays a similar role to the mapping specification in MDA. The proposed solution to simplify the determination of these correspondences in MDA is based on the use of a semi-automatic matching tool, i.e., automatically detecting what elements are corresponding and interacting with a user so as to validate the suggestions made by the *matching algorithm* [6].

In this paper the problem of determining mapping specification is investigated. An algorithm for detecting correspondences presented in [7] and applied to *schema matching* in the field of database systems is enhanced and adapted to be used in field of MDA. An implementation of this algorithm was done using a tool called Semi-Automatic Tool for Model-Driven Engineering (SAT4MDE) [6].

This paper is organized as follows. Section 2 presents an overview of the technologies and concepts involved in the software development process using a model driven approach. Section 3 presents an approach and an algorithm to make the semi-automatic mapping specification possible. Section 4 describes the tool SAM4MDE and its extension to insert the algorithm to generate mapping specification. Section 5 presents tests and the evaluation of the results obtained by the semi-automatic generation of mapping specification. Section 6 contains conclusions and the future directions of this research work.

2 Background

The aim of this section is to present some technologies and tools used in the context of Model-Driven Engineering (MDE) and consequently related them to our research work.

2.1 Model Driven Engineering

Model Driven Engineering (MDE) is an approach that has models as main focus in order to provide benefits such as cost reduction and increased quality of software products. The relevance of models in MDE does not only consist of documenting software systems, but these are formal models that can be understood by computers, i.e. they contain information that can be easily manipulated by a computer. The model manipulation is done through model transformation, a technique that consists in obtaining models from another model. Any modification made in the software is done inside models and the transformation is repeated so as to diffuse the changes.

Model-Driven Engineering is a recent approach and requires sophisticated formalism, i.e. stable technique and tools that allow the creation of consistent software, following a methodology for applying MDE. Some initiatives have been proposed in the

recent years, e.g. *Model Driven Architectures* (MDA) from *Object Management Group* (OMG), *Eclipse Modeling Framework* from *Eclipse Project* and *Software Factories* from *Microsoft*.

2.2 Mapping Specification and Transformation Definition

Figure 1 illustrates our approach for generating transformation definition from mapping specification. The mapping model (i.e. mapping specification) contains the correspondences between the source metamodel (*left*) and the target metamodel (*right*). A *transformation program* is based on a *transformation model* that is generated from the *mapping specification*. After this, the *target model* is created through the execution of a *program transformation* by the *transformation engine* that takes as input the *source model*, the *source metamodel* and the *target metamodel*.

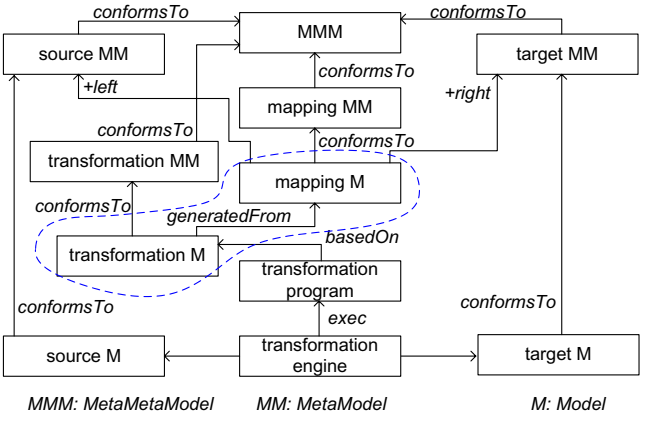


Fig. 1. An approch for MDE [8]

2.3 Tools for Model Matching

Nowadays many tools have been developed with the objective to help the developer in the process of creating the *schema matching* [6][8] [9]. These tools are developed using algorithms that detect similarities between metamodels (in database, the metamodels are database schema). In general, tools for matching metamodels (i.e. schema) are more common in database field. Clio [9][10], Tess [11] and SAMT4MDE [12] are examples of tools for metamodel matching.

3 Proposed Approach for Metamodel Matching

The proposal is to develop semi-automatic tasks, i.e. use an approach that searches for similarities between elements of involved metamodels.

3.1 Foundation for Metamodel Matching

A transformation function can be defined as follows:

$Transf(M_1(s)/M_a, C_{Ma \rightarrow Mb}/M_c) \rightarrow M_2(s)/M_b$ [12]

where:

M_1 is a model for a system s created using the metamodel M_a,

M_2 is a model of the same system s created using the metamodel M_b,

$C_{Ma \rightarrow Mb}$ is the mapping model between M_a and M_b created using the metamodel M_c.

The research work presented in this paper is based on the mathematical model and definitions from [12] that presents the operator $Match(Ma, Mb) = C_{Ma \rightarrow Mb}$ that takes two metamodels as input and produces a mapping model between them, and being this mapping model conform to a mapping metamodel.

3.2 Another Algorithm for Metamodel Matching

In [12], an approach for metamodel matching is presented and its match operator implements an algorithm based on *cross relationships* [13], comparisons between classes, data types and enumerations. The selection of equal or similar classes, data types and enumerations are achieved by a function φ that returns a discrete value for each pair compared: 1 (one) if the classes or data types or enumerations are equals; 0 (zero) if the classes or data type or enumerations are similar; -1 (minus one) if the clasess or data types or enumerations are different.

In this paper, our contribution to this field is another algorithm that uses structural comparison between a class and its neighbor classes in order to select the equal or similar classes from source and target metamodel. The proposed algorithm for *metamodel matching* is an extension and enhancement of the algorithm presented in [7] and it is implemented in the Semi-Automatic Matching Tool for MDE (SAMT4MDE).

The similarity function between two classes $c1$ and $c2$ is given by:

$$similarity(\text{c1,c2}) = basicSim(\text{c1,c2}) * coefBase + structSim(\text{c1,c2}) * coefStruct$$

where $0 \leq coefBase \leq 1$, $0 \leq coefStruct \leq 1$, and $coefBase + coefStruct = 1$.

The function $similarity$(c1,c2) is the weighted mean which has as parameters $basicSim(c1, c2)$ and $structSim$(c1,c2) with the weights $coefBase$ and $coefStruct$, respectively. It returns continuous values representing the similarity level between $c1$ and $c2$. On the contrary the algorithm presented in [12] that returns discrete values being each value associated to a result (e.g., value 1 means that classes are equal), the similarity function presented here returns continuous values, because the similarity level between two classes is not obtained by only one deterministic value, but a range of possible solutions (e.g. values between 0.7 and 1 means classes are equal).

The weights are $coefBase$ and $coefStruct$ that added result in 100%, or 1(one). For example, if $coefBase$ is equal to 0.3, then $coefStruct$ must be equal to 0.7. The value of $similarity$(c1,c2) is compared to a $threshold$ value in the range [0,1]. If a similarity is greater than $threshold$ value, the classes $c1$ and $c2$ are correspondent, otherwise they are not correspondent. Thus, $threshold$ is an important point to take

a decision: if the value is low, in general in the range [0, 0.5], many elements will be considered correspondent in a wrong way (false positive), if the value is high, in general in the range [0.8, 1], many classes will not be considered as correpondent (false negative). In our experiments, we have used the $threshold = 0.6$.

The function $basicSim(c1, c2)$ compares the classes $c1$ and $c2$ based on a repository of taxonomies. It is similar to the function φ. However, $basicSim(c1, c2)$ returns values in the range [0,1].

The function $structSim(c1, c2)$ is based on the structural similarity between the classes $c1$ and $c2$. The structural neighbors of a class C constitute a quadruple:

<ancestor(C), sibling(C), immediateChild(C), leaf(C)>,

where:

- ancestor(C): is a set of classes that are fathers of C, from the root element until the immediate father of C.
- sibling(C): is a set of classes that shares the same immediate father of class C.
- immediateChild(C): is a set of classes that are direct descendants of class C.
- leaf(C): is a set of leaf classes from the sub-tree that has the class C as root.

The set of classes that constitutes the quadruple of structural neighbors are selected following determined criteria. The *ancestor elements* influences its descendants. However, two classes can share the same structure of ancestors, however, they can be different in the structure of *sibling*. Futhermore, to consider the structural details of a class, the *immediateChild* are analyzed and the last level of descendants of the class, i.e. *leaf*. Figure 2 illustrates a class C and its structural neighbors.

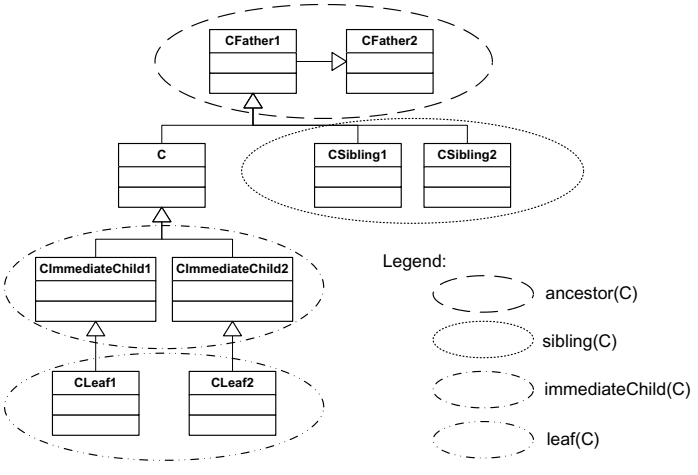


Fig. 2. Structural neighbors of a class C

To calculate the structural similarity between two classes $c1$ and $c2$, the structural neighbors of $c1$ and $c2$ that are denominated $V(c1)$ and $V(c2)$ must be obtained. The neighbors are:

$V(c1)$=< $ancestor(c1), sibling(c1), immediateChild(c1), leaf(c1)$ >
$V(c2)$=< $ancestor(c2), sibling(c2), immediateChild(c2), leaf(c2)$ >

The structural similarity is obtained in function of partial similarities as follows:

- $ancestorSimClass(c1, c2)$: calculates the similarity between ancestors of $c1$ and $c2$.
- $siblingSimClass(c1, c2)$: calculates the similarity between the sibling of $c1$ and $c2$.
- $immediateChildSimClass(c1, c2)$: calculates the similarity between the immediate sibling of $c1$ and $c2$.
- $leafSimClass(c1, c2)$: calculates the similarity between leaves from a subtree whose root is $c1$ and $c2$.

Each function populates an array *M* with dimensions *m1 x m2*, with *m1* the size of the set of classes related to $c1$, and *m2* with the size of the set of classes related to $c2$. For example, each function $ancestorSimClass(c1, c2)$, *m1* is an amount of classes ancestors of $c1$ and *m2* is the amount of classes ancestor of $c2$. Listing 1.1 presents the algorithm to construct the array *M* for the function $ancestorSimClass(c1, c2)$. Similarly, the other functions constitute an array *M*.

Listing 1.1. Algorithm for determining similarity between pairs of ancestors.

```
1 for i = 0 until size of (ancestor(c1)) do
    for j = 0 until size of (ancestor(c2)) do
3    M[i][j] <- basicSim (ancestor(c1).get(i), ancestor(c2).get(j));
    endfor;
5 endfor;
```

Figure 3 represents an array M obtained through the execution of algorithm presented in Listing 1.1. Each row represents an ancestor class of $c1$ and each column represents an ancestor class of $c2$. The value 0.7 for the row $R0$ and the column $C1$ means that ancestor class $R0$ of $c1$ has a similarity value of 0.7 in relationship to ancestor class $C1$ of $c2$.

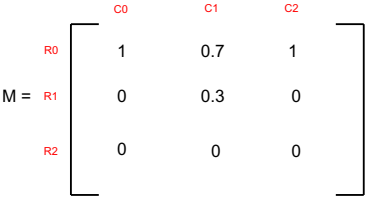


Fig. 3. An array created by the algorithm of Listing 1.1

After the creation of *M*, each function of partial similarity invokes a function *agg* that has the array *M* and a value of *threshold* as parameters. The value of *threshold* is in the range [0,1]. A function *agg* calculates the average (*avg*), the standard deviation (*sd*) and the variance coefficient (*vc*) of an array M. For a function $ancestorSimClass(c1,$

$c2$), these measures are given by the following equations (for the other functions, the same principle is applicable):

$$avg = \frac{\sum_{i=1}^{|ancestor(c1)|} \left(\sum_{j=1}^{|ancestor(c2)|} M[i][j] \right)}{|ancestor(c1)| * |ancestor(c2)|} \text{ (Eq. 1)}$$

$$sd = \frac{\sum_{i=1}^{|ancestor(c1)|} \left(\sum_{j=1}^{|ancestor(c2)|} M[i][j] - avg)^2 \right)}{|ancestor(c1)| * |ancestor(c2)|} \text{ (Eq. 2)}$$

$$vc = \frac{sd}{avg} \text{ (Eq. 3)}$$

When calculating these measures, the function *agg* results in values as follows:

- If $threshold \geq vc$, the coefficient of variance is below the established limit, then the average can be considered a reliable value, and the function *agg* returns an average value (avg).
- If $threshold < vc$, the function *agg* excludes the values of M that are below $avg * (1 - threshold)$, because these values are contributing to dispersion of the average, and calculates news measures. This procedure continues until $threshold \geq vc$. Listing 1.2 presents this function *agg*.

Listing 1.2. Representation of function *agg*.

```
1 float agg(float m[][], float thr, float lastvl) {
        float average = avg(m, lastvl);
3       float standardDeviation = sd(m, lastvl);
        float vc = (float) (standardDeviation / average);
5       float result = 0;
        if (thr >= vc) { return average;} else {
7               float lastvl = average * (1 - thr);
                result = agg (m, thr, lastvl);}
9       return result;
  }
```

The value of the function *agg* is used by the function of partial similarity. Each of the four functions uses the function *agg* to determine the similarity result. After the four partial similarities are obtained ($ancestorSimClass(c1, c2)$, $siblingSimClass(c1, c2)$, $immediateChildSimClass(c1, c2)$ and $leafSimClass(c1, c2)$), the whole structural similarity can be calculated. Function $structSim(c1, c2)$ is given by:

$$structSim(c1, c2) = ancestorSimClass(c1, c2) * coefAnc + siblingSimClass(c1, c2) * coefSib + immediateChildSimClass(c1, c2) * coefImmC + leafSimClass(c1, c2) * coefLeaf$$

Where, $0 \leq coefAnc \leq 1$,
$0 \leq coefSib \leq 1$,
$0 \leq coefImmC \leq 1$,
$0 \leq coefAnc \leq 1$,
and $coefAnc + coefSib + coefImmC + coefLeaf = 1$.

The distribution between the coefficients must be uniform in order to compute the structural similarity in a flexible and complete manner. For this purpose all structural neighbors of an element are considered important. Therefore, each partial similarity has a contribution with a similar weight to the weight of another parts.

4 Extending and Adapting the SAMT4MDE

The solution for semi-automatic mapping generation developed in this work is built by extending and adapting the Semi-Automatic Matching Tool for MDE (SAMT4MDE) [6]. Due to its extensibility, SAMT4MDE enables developers to create search engines for correspondences, and attach this engine to the tool.

This Section presents implementation aspects about the algorithm detailed in Section 3.2 and the interaction with the tool SAMT4MDE.

4.1 Modeling

The tool SAMT4MDE initially presented in [12] is used as base in order to code the algorithm for searching structural similarities presented in Section 3.2. This tool is implemented using an EMF framework. Figure 4 presents a simplified class diagram that demonstrates the main functionalities of this tool.

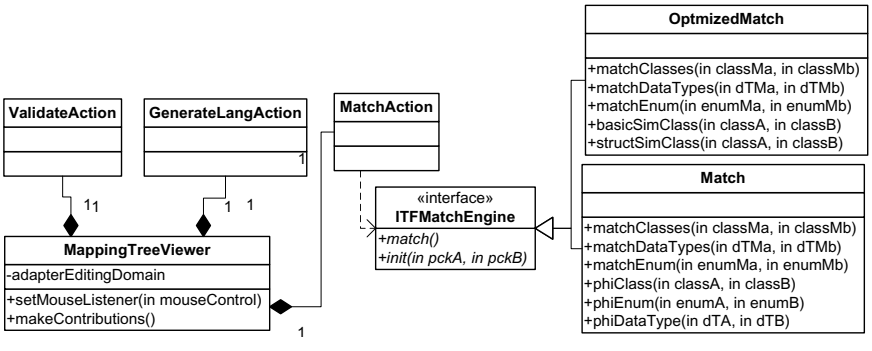


Fig. 4. Simplified class diagram for SAMT4MDE

The class *MappingTreeViewer* handles mappings using trees, and controls mapping updates through *adapterEditingDomain* attribute. The method *setMouseListener* registers the class as a listener to "listen" to mouse click events, and the method *makeContributions* creates the objects that execute tool actions.

The following classes represent tool actions: *ValidateAction* that contains code to validate mapping, *MatchAction* that contains code to execute semi-automatic metamodel matching and *GenerateLangAction* that contains code to generate transformation definition written in a specific transformation language. The class *MatchAction* is in charge of containing the metamodel matching action.

The class *MatchAction* contains attributes and methods which permit interaction with classes implementing metamodel matching. The method *run* invokes *init* method,

which receives packages from source and target metamodels in order to provide metamodel paths to the class implementing *ITFMatchEngine* interface. This class therefore presents a behavior that enables navigation inside the metamodel. *Match* and *OptmizedMatch* classes implement *ITFMatchEngine*, so they represent two different implementations. *Match* class is depicted in [12], and *OptmizedMatch* class implements an algorithm for searching structural similarities presented here. This tool allows the user to choose what implementation to run.

After calling *init* method, *MatchAction* object invokes *match* method which runs the algorithm for searching structural similarities. For taking *match* action, *basicSimClass* and *structSimClass* methods presented in Section 3.2 are used.

The sequence diagram to generate semi-automatic mapping specification is presented in Figure 5.

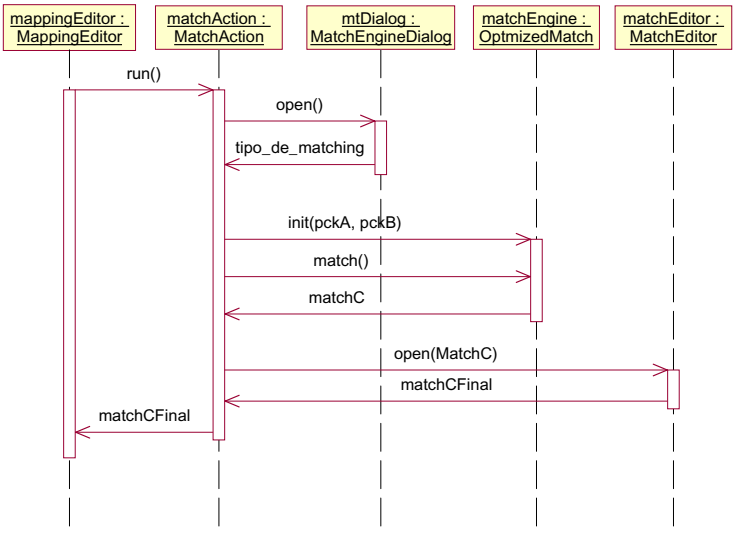


Fig. 5. Sequence diagram for use case *Generate mapping specification*

MappingEditor is an object that represents the user interface. This object invokes *run()*, a *matchAction*'s method that contains the action of generating mapping specification. The object *matchAction* calls the method *open()* in *MatchEngineDialog* class, which is responsible for allowing the user to choose what type of metamodel matching is to be executed.

After choosing what metamodel matching is going to be executed, the object *matchAction* forwards to object *matchEngine* the packages that contain the metamodels, through the method *init(pckA, pckB)*. After this, *matchAction* invokes *match()*, which executes the algorithm described in Section 3.2. The object *matchEngine* returns correspondences (*matchC*) to *matchAction* before user validation. In the next step, *matchAction* invokes the method *open(matchC)* which forwards to *matchEditor* the mapping specification. *MatchEditor* is an interface object with a window that allows the user to validate the desired correspondences. As soon as the user validates correspondences

matchEditor produces the final mapping specification, i.e. (*matchCFinal*), which is forwarded to user interface object.

4.2 Prototyping

SAMT4MDE is a plug-in for Eclipse and it can interact with Mapping Tool for MDE (MT4MDE) that is another plug-in that supports the manual creation and edition of mapping specification.

The MT4MDE graphic user interface (GUI) is implemented as a plug-in for Eclipse in which source and target metamodels are loaded in the left and right panel, respectively, and the mapping model lies in center.

Figure 6 shows the correspondences generated by SAMT4MDE. Subsequently, a user can validate the automatic correspondences generated by the proposed algorithm based on structural similarities.

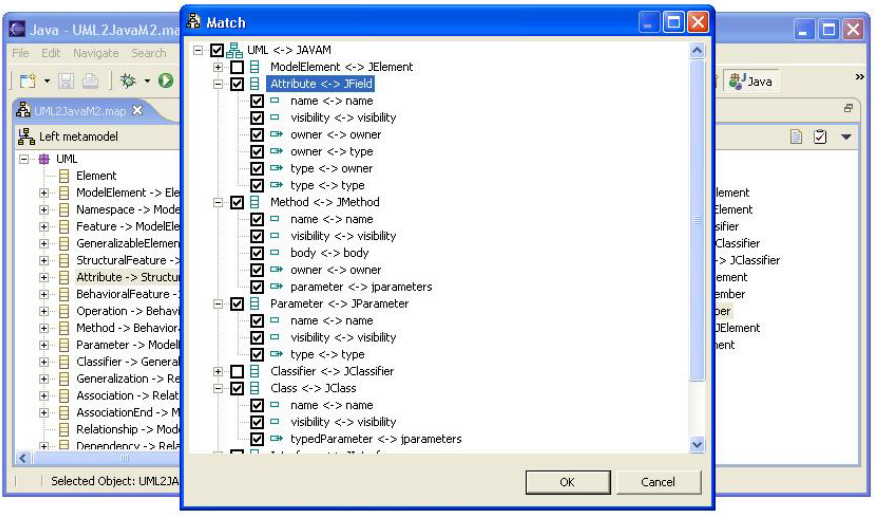


Fig. 6. Elements matched by SAMT4MDE in validation process

After mapping validation, SAMT4MDE pass the mapping specification to MT4MDE and a user can edit it. Figure 7 illustrates this moment. New correspondences can be added to the model by user in this manner.

5 Tests

A test case for creating mapping specification between UML and Java metamodels is proposed in order to evaluate the algorithm developed in this paper for metamodel matching. Some quality measures are calculated as a way to analyze the quality of obtained results in test cases of this type. In this paper, we have used the quality measures presented in [14].

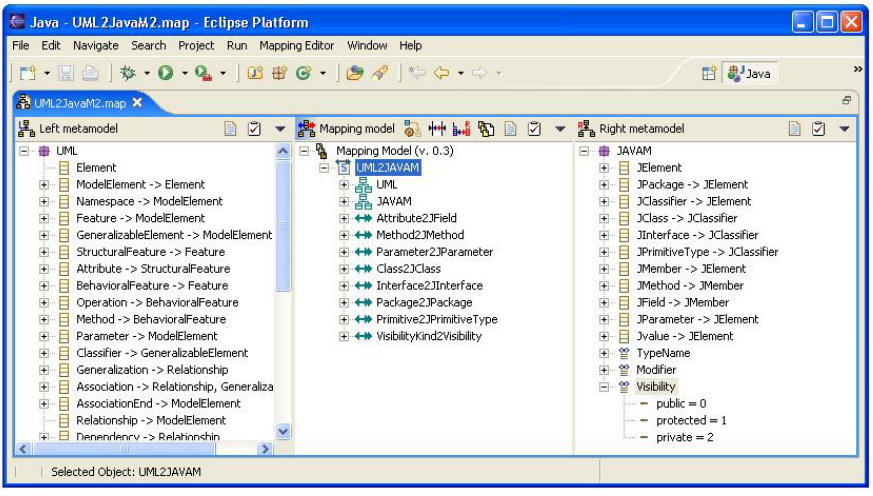


Fig. 7. Mapping Specification validated by user

The SAMT4MDE produced the following results for this study case: Schema similarity = 0.68, Precision = 0.84, Recall = 0.90, F-Measure = 0.87 and Overall = 0.73.

The similarity between UML and Java metamodels is 0.68. It means that 68% of elements from both metamodels are involved in metamodel matching. A high percentage to metamodel similarity means that semantic distance between these metamodels is small, and a low percentage means the opposite.

The measure of precision is 0.84. It is assumed from this information that 84% of found correspondences are correct. The measure of recall is 0.90, meaning that 90% of existing correspondences were found.

6 Conclusions

The main contribution of this paper is to present an algorithm for metamodel matching and its implementation in a tool which is capable of semi-automatically creating mapping specifications, making matching suggestions that can be evaluated by users. This provides more reliability to the system because mapping becomes less error-prone.

The algorithm proposed can identify structural similarities between metamodel elements. However, sometimes elements are matched by its structures but they do not share their meanings. The lack of analysis about element meaning leads the tool to find false positives, i.e. derived correspondences that are not real.

Future works aim to minimize the appearance of errors (i.e. false positives and false negatives) in automatic correspondences by using semantic analysis techniques and machine learning. The semantic analysis would help to analyze element meanings before the tool infers element matches, and machine learning would create a mechanism for progressive improvement of the mapping generation task from learning previous mappings.

Acknowledgements. This research work is supported by CNPq and FAPEMA.

References

1. Jouault, F., Kurtev, I.: On the Architectural Alignment of ATL and QVT. In: SAC 2006: Proceedings of the 2006 ACM symposium on Applied computing, pp. 1188–1195. ACM Press, New York (2006)
2. Muliawan, O.: Extending a Model Transformation Language using Higher Order Transformations. In: IEEE 15th Working Conference on Reverse Engineering, pp. 315–318 (2008)
3. OMG: Meta Object Facility (MOF) 2.0 Query/View/Transformation Specification - Final Adopted Specification, ptc/07-07-07 (2007)
4. Patrascoiu, O.: Mapping EDOC to Web Services using YATL. In: 8th IEEE International Enterprise Distributed Object Computing Conference (EDOC 2004), pp. 286–297 (2004)
5. Sims, O.: Enterprise MDA or How Enterprise Systems Will Be Built, MDA Journal, Meghan Kiffer Pr (2004)
6. Lopes, D., Hammoudi, S., Sousa Jr., G., Bontempo, A.: Metamodel Matching: experiments and comparison. In: IEEE International Conference on Software Engineering Advances (ICSEA 2006) (2006)
7. Chukmol, U., Rifaiem, R., Benharkat, N.: EXSMAL: EDI/XML Semi-Automatic Schema Matching ALgorithm. In: Proceedings of the Seventh IEEE International Conference on E-Commerce Technology, pp. 422–425. IEEE Computer Society, Los Alamitos (2005)
8. Lopes, D.: Study and Applications of the MDA Approach in Web Service Platforms, Ph.D. thesis (written in French), University of Nantes (2005)
9. Popa, L., Velegrakis, Y., Miller, R.J., Hernandez, M., Fagin, R.: Mapping Generation and Data Translation of Heterogeneous Web Data. In: International Workshop on Data Integration over the Web (DIWeb) (2002)
10. Hernandez, M.A., Ho, H., Popa, L., Fukuda, T., Fuxman, A., Miller, R.J., Papotti, P.: Creating Nested Mappings with Clio. In: IEEE 23rd International Conference on Data Engineering (ICDE), pp. 1487–1488 (2007)
11. Lerner, B.S.: A Model for Compound Type Changes Encountered in Schema Evolution. In: ACM Transactions on Database Systems (TODS), vol. 25(1), pp. 83–127. ACM Press, New York (2000)
12. Lopes, D., Hammoudi, S., Abdelouahab, Z.: Schema Matching in the Context of Model Driven Engineering: From Theory to Practice. In: Proceedings of the International Conference on Systems, Computing Sciences and Software Engineering (SCSS 2005), pp. 219–227 (2005)
13. Pottinger, R.A., Bernstein, P.A.: Merging Models Based on Given Correspondences. In: Proceedings of the 29th VLDB Conference, pp. 826–873 (2003)
14. Do, H., Melnik, S., Rahm, E.: Comparison of Schema Matching Evaluations. In: Revised Papers from the NODe 2002 Web and Database-Related Workshops on Web, Web-Services, and Database Systems, pp. 221–237. IEEE Computer Society, Los Alamitos (2003)

A Study of Indexing Strategies for Hybrid Data Spaces

Changqing Chen[1], Sakti Pramanik[1], Qiang Zhu[2], and Gang Qian[3]

[1] Department of Computer Science and Engineering
Michigan State University, East Lansing, MI 48824, USA
chencha3@cse.msu.edu, pramanik@cse.msu.edu
[2] Department of Computer and Information Science
The University of Michigan - Dearborn, Dearborn, MI 48128, USA
qzhu@umich.edu
[3] Department of Computer Science
University of Central Oklahoma, Edmond, OK 73034, USA
gqian@uco.edu

Abstract. Different indexing techniques have been proposed to index either the continuous data space (CDS) or the non-ordered discrete data space (NDDS). However, modern database applications sometimes require indexing the hybrid data space (HDS), which involves both continuous and non-ordered discrete subspaces. In this paper, the structure and heuristics of the ND-tree, which is a recently-proposed indexing technique for NDDSs, are first extended to the HDS. A novel power value adjustment strategy is then used to make the continuous and discrete dimensions comparable and controllable in the HDS. An estimation model is developed to predict the box query performance of the hybrid indexing. Our experimental results show that the original ND-tree's heuristics are effective in supporting efficient box queries in the hybrid data space, and could be further improved with our proposed strategies to address the unique characteristics of the HDS.

Keywords: Hybrid data space, Database, Access method, Multidimensional indexing, Box query.

1 Introduction

In many contemporary database applications, indexing of hybrid data which contains both continuous and discrete dimensions is required. For example, when indexing weather data of different locations, the daily temperature, precipitation, humidity should be treated as continuous information while other information such as the type of precipitation is typically regarded as discrete. Different indexing techniques have been proposed for either the CDS or the NDDS. Examples for CDSs indexing methods are the R-tree [6], R*-tree [1], K-D-B-tree [12] and LSDh-tree [7]. NDDSs indexing techniques include the ND-tree [9,10] and the NSP-tree [11]. Not surprisingly, all these indexing methods could not be applied to the HDS directly because they rely on domain-specific characteristics (e.g., the order of data in the CDS) of their own data spaces.

One way of applying the CDS/NDDS indexing techniques to the HDS is to transform data from one space to the other. For example, discretization methods [2,5,8] could be

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 149–159, 2009.
© Springer-Verlag Berlin Heidelberg 2009

utilized to convert data from continuous space to discrete space. If we discretize the weather data mentioned before, daily temperature could be converted to three discrete values: cold, warm and hot. However, this approach clearly changes the semantics of the original data.

The C-ND tree [4] is recently proposed to create indexes for the hybrid data space, and is optimized to support range queries in the HDS. In this paper, we evaluate the effectiveness of the extended ND-tree structure and heuristics for box queries in the HDS. The ND-tree structure and building algorithms/heuristics are extended to handle both continuous and discrete information of the HDS. A novel strategy using power adjustment values to balance the preference for continuous and discrete dimensions is presented to handle the unique characteristics of the HDS. And an effective cost model to predict the performance of HDSs indexing is introduced. Our experimental results show that, the extended heuristics are effective in supporting box queries in the HDS, and the cost estimates from the presented performance model are quite accurate.

The rest of the paper is organized as follows. In Section 2 the ND-tree data structures and heuristics are extended to the HDS, and an approach of using different power (exponent) values to handle the unique characteristics of the HDS is presented. Section 3 reports our experimental results, which demonstrate that hybrid space indexing is quite promising in supporting efficient box queries in HDSs. Section 4 outlines a model to predict the performance of hybrid space indexing. Section 5 describes the conclusions and future work.

2 The Extended Hybrid Indexing

2.1 Hybrid Geometric Concepts and Normalization

To efficiently build indexes for the HDS, some geometric concepts need to be extended from the NDDS to the HDS. The hybrid geometric concepts used in this paper are the *hybrid (hyper-)rectangle* in the HDS, the *edge length* of a hybrid rectangle on a given dimension, the *area* of a hybrid rectangle, the *overlap* between two hybrid rectangles and the *hybrid minimum bounding rectangle (HMBR)* of a set of hybrid rectangles. Detailed definition of these concepts could be found in [4] and is omitted here due to page limit.

One challenge in applying hybrid geometric concepts is how to make the measures for the discrete and continuous dimensions comparable. For example, how to compare the size of 2 for a discrete component set (i.e., 2 letters/elements) of an HMBR with the length of 500 for a continuous component interval in the same HMBR? To solve the problem, we adopt normalized measures for hybrid geometric concepts introduced in [4]. In the rest of this paper, we always use normalized hybrid geometric measures unless stated otherwise.

2.2 Extending the ND-Tree to the HDS

Each non-leaf node entry in the hybrid indexing tree keeps an HMBR of a child node and a pointer to that child node. Information for discrete dimensions in the HMBR

is stored using a bitmap representation and information for continuous dimensions is stored by recording the corresponding lower and upper bounds of each dimension. Each leaf node entry stores the discrete and continuous component of every dimension as well as a pointer pointing to the actual data associated with the key in the database.

Two critical tasks, namely choosing a leaf node to insert a new vector and overflow treatment are extended to the HDS, by using the corresponding geometric concepts defined in Section 2.1. When splitting an overflowing node, sorted entry lists are generated for a continuous dimension C by sorting all entries' lower bound values and then upper bound values on C. Detailed discussion of these two tasks is omitted in this paper and could be found in [9,10].

Given the extended insert operation in the HDS, the delete operation is implemented as follows. If no underflow occurs after an entry is removed from a leaf node, only the ancestor nodes' HMBRs are adjusted. In case of an underflow, the whole node is removed and all the remaining entries in that node are reinserted.

A query box in the HDS is a hybrid rectangle containing a query range/set on each dimension. For a continuous dimension, a query range is specified by its upper and lower bounds. For a discrete dimension, a query set is specified by a subset of letters/elements from its domain. A traditional depth-first search algorithm is implemented for the hybrid indexing tree and the details are omitted in this paper.

2.3 Enhanced Strategy for Prioritizing Discrete/Continuous Dimensions

As mentioned in Section 2.1, to make a fare comparison between discrete and continuous dimensions, we have employed normalized edge lengths when calculating geometric measures for an HDS. This strategy allows each dimension to make suitable contributions relative to its domain size in the HDS.

We also notice that the (non-ordered) discrete and continuous dimensions usually have different impacts on the performance of hybrid indexing due to their different domain properties. For example, a discrete dimension is more flexible when splitting its corresponding component set of an HMBR due to the non-ordering property, resulting in a higher chance to obtain a better (smaller overlap) partition of the relevant overflow node. Assume $S_1 = \{a, g, t, c\}$ is the component set of an HMBR on a discrete dimension, the letters in S_1 can be combined into two groups arbitrarily (subject to the minimum space utilization constraint of the node) to form a split of the set, e.g., $\{g\}/\{a, t, c\}$, $\{a, t\}/\{g, c\}$, etc. On the other hand, for the component set on a continuous dimension, say $S_2 = [0.2, 0.7]$, the way to distribute a value in S_2 totally depends on the splitting point. If the splitting point is 0.5 (i.e., having a split $[0.2, 0.5]/(0.5, 0.7]$), the values less than or equal to 0.5 have to be in the first group, and the others belong to the second group, because of the ordering property of a continuous dimension. The challenge is how to make use of this observation to balance the preference for discrete and continuous dimensions in the HDS.

One might suggest adopting different weights for the (normalized) edge lengths of an HMBR on the discrete and continuous dimensions, respectively. Unfortunately, this approach can not work. Two important measures used in the tree construction algorithms are the area of an HMBR (or overlap between HMBRs) and the span of an HMBR on a particular dimension (i.e., the edge length of the HMBR on the given dimension).

Suppose we have HMBRs (or overlaps) R_1 and R_2 in a two-dimensional HDS with the following relationship: $Area(R_1) = L_{11} \times L_{12} < Area(R_2) = L_{21} \times L_{22}$, where L_{ij} is the (normalized) edge length of R_i on the *j-th* dimension ($j = 1, 2$). Assume that the first dimension is discrete and the second one is continuous. If we assign a weight w_d to the discrete edge length and another weight w_c to the continuous edge length when calculating the area, we will still have $Area(R1) = (w_d \times L_{11}) \times (w_c \times L_{12}) < Area(R2) = (w_d \times L_{21}) \times (w_c \times L_{22})$ because the weight factors on both sides of the inequality cancel each other. The same observation can be obtained for spans.

To overcome the above problem, we adopt another approach by assigning different power (exponent) values p_d and p_c to the discrete and continuous edge lengths, respectively, when calculating area values. With a normalization, we can assume $p_c = (1 - p_d)$. For R_1 and R_2 in the above example, if$L_{11} = 0.1, L_{12} = 0.3, L_{21} = 0.2, L_{22} = 0.2, p_c = 0.1, p_d = 0.9$, we have $Area(R_1) = L_{11}^{p_d} \times L_{12}^{p_c} = 0.1^{0.1} \times 0.3^{0.9} \approx 0.27 > Area(R_2) = L_{21}^{p_d} \times L_{22}^{p_c} = 0.2^{0.1} \times 0.2^{0.9} \approx 0.20$, while the original area values have the relationship $Area(R_1) = 0.1 \times 0.3 = 0.3 < Area(R_2) = 0.2 \times 0.2 = 0.4$. Hence, we can change the area comparison result by using different power adjustment values. Since the edge length is normalized to be between 0 and 1, the larger the power value is, the smaller the adjusted length would be (unless the edge length is 1 or 0). For the heuristics involving areas comparison during tree construction, we always prefer a smaller area. Therefore, if we increase power p_d (i.e., reduce p_c) for discrete dimensions, we make the discrete edge lengths contribute less to the area calculation while making the continuous edge lengths contribute more to the area calculation. In this sense, we make the discrete dimensions more preferred. The way to make the continuous dimensions more preferred is similar.

We can also assign power adjustment values q_d and $q_c = (1 - q_d)$ to the discrete and continuous edge lengths, respectively, when calculating the span value for every dimension. However, during tree construction time a dimension with a larger span is more preferred. Therefore, if we want to make a discrete dimension more preferable, we need to decrease power value q_d for that dimension, which is different from the situation of calculating areas values. The discussion for the span value on continuous dimensions is similar. If we want to make discrete dimensions consistently more preferred during the tree construction, we need to increase p_d and, in the meantime, decrease q_d.

To reduce the number of parameters for the algorithm, we simply let $q_d = (1 - p_d)$. Hence, we only need to set a value for one parameter p_d , other parameters (i.e., $p_c,\ q_d,\ q_c$) are generated according to p_d. The experimental results in Section 3 show that this power adjustment strategy further improves the performance of the hybrid indexing.

3 Experimental Results

3.1 Experimental Setup

Data sets used for our experiments were randomly generated which consist both continuous and discrete dimensions. For a discrete dimension if the alphabet size is A, a discrete value was created by generating a random integer between 0 and $A - 1$. For a

continuous dimension if the range is A, the possible values are decimal numbers ranging between 0 and A. For a test query, a box size X is used to define the volume of its query box. Given a query box with box size X, each discrete dimension has X letters and each continuous dimension has length X. The query performance is measured by the number of I/Os (i.e., the number of index tree nodes accessed, assuming each node occupies one disk block) and is computed by averaging the I/Os over 200 queries.

In the following subsections we compare performances of the hybrid indexing tree, the ND-tree, the R*-tree and the 10% liner scan[3]. For the same reason discussed in [4], we keep both continuous and discrete data in the leaf nodes of the ND-tree and the R* tree. Various parameters such as database sizes and alphabet sizes are considered in our experiments. A symbol δ is used to represent the additional dimensions utilized by the hybrid indexing approach. For example, given a HDS with i continuous dimensions and j discrete dimensions, by indexing the whole HDS we have $\delta(\delta = j)$ extra dimensions to use when compared to the R*-tree approach, and $\delta(\delta = i)$ extra dimensions to use when compared to the ND-tree approach.

In our experiments we create the R*-tree for a 4-dimensional continuous subspace, which is a typical dimension number for the R*-tree to avoid the dimensionality curse problem. For the discrete subspace we use 8 dimensions because an effective ND-tree could not be built if the number of dimensions is too low (there are too many duplicate vectors in the subspace).

From the experiment results we see that the hybrid indexing outperforms the other three approaches. In some of the cases, the performance gain is quite significant.

3.2 Performance Gain with Increasing Database Sizes

In this group of tests the performance of hybrid indexing is compared with that of the ND-tree, the R*-tree and the 10% linear scan for various database sizes (i.e., the number of vectors indexed). The number of additional dimensions δ is set to 2. That is, we use the hybrid indexing approach to index the 4 continuous dimensions used by the R*-tree plus 2 additional discrete dimensions, and compare the query I/O with that of the R*-tree. Similarly, we compare the performance of indexing 8 discrete dimensions and 2 continuous dimensions against the ND-tree approach which indexes only the 8 discrete dimensions. The query I/O of hybrid indexing is also compared with that of the 10% linear scan approach, which utilizes all the dimensions as the hybrid indexing does. The alphabet size for each of the discrete dimensions is set to 10. Figure 1 shows that the hybrid indexing approach reduces box query I/Os and the performance gain generally increases with growing database sizes.

3.3 Performance for Various Additional Dimensions

In the following experiments, we varies the number of additional dimensions (i.e., the δ value) used by the hybrid indexing. The alphabet size and database size in these experiments are set to 10 and 10 million respectively. Our experimental results are reported in Figure 2. Again from these results we see that the hybrid indexing outperforms all the other approaches. In some cases (e.g., compared with the R*-tree) the performance improvement is significant.

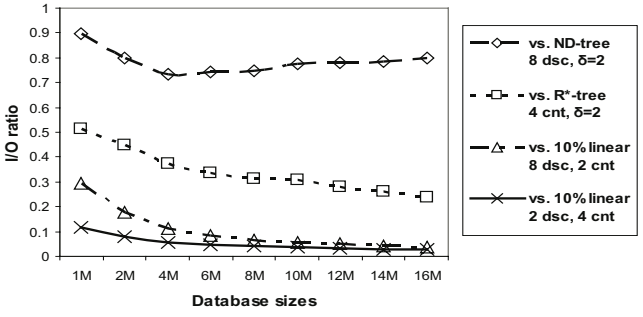


Fig. 1. Effect of various database sizes

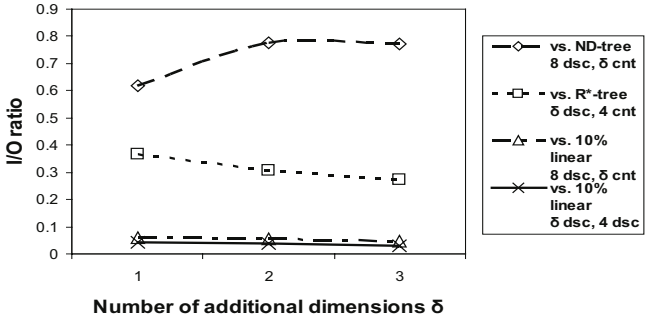


Fig. 2. Effect of additional dimensions

3.4 Performance for Different Alphabet Sizes

As we see from Figure 3, the hybrid indexing is much more efficient when compared to the R*-tree and the 10% linear scan. It also does better than the ND-tree approach. With increasing alphabet sizes the ND-tree performance gets closer to the hybrid indexing. However, in real world applications most NDDS domains are small. For example, genome data has a domain size of 4 (i.e., $\{a,\ g,\ t,\ c\}$). The database size used for this group of tests is 10 million.

3.5 Performance for Different Query Box Sizes

All of the above experiments show the performance comparisons for query box size 2. This group of tests evaluates the effect of different query box sizes. Query results for box size $1 \sim 3$ are reported here because as box sizes become larger, the 10% linear scan approach is more preferable. Not surprisingly, all indexing trees will eventually lose to linear scan when the query selectivity is high. The results in Figure 4 show that indexing the hybrid data space increases box query performance for all the box sizes given. Database size 10 million and alphabet size 10 are used for this group of tests.

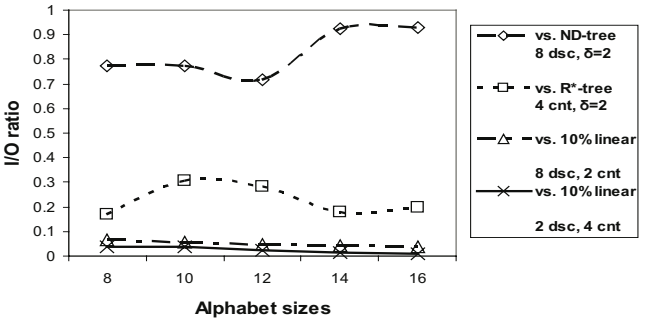


Fig. 3. Effect of different alphabet sizes

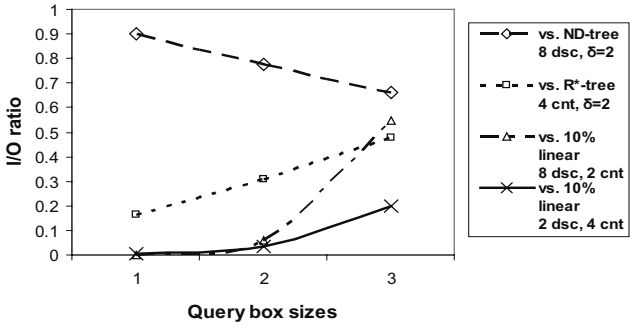


Fig. 4. Effect of different query box sizes

3.6 Effect of Enhanced Strategy with Power Value Adjustment

To examine the effectiveness of using exponent (power) values to adjust the edge lengths of an HMBR, as discussed in Section 2.3, we conducted relevant experiments. The query I/Os of using this enhanced strategy are shown in Figure 5. The x-axis indicates different power values (p_d) used for discrete dimensions. To eliminate the possible effect that different number of discrete and continuous dimensions might have on the enhanced strategy, we use an HDS with 4 discrete and 4 continuous dimensions. The alphabet size is set to 10 and number of vectors indexed is 10 million. The results in Figure 5 show that the new enhanced strategy could further improve the performance of the hybrid indexing using extended ND-tree heuristics. The exponent value of 0.5 corresponds to the situation of not applying any power value adjustment because both discrete and continuous dimensions have the same exponent value (0.5).

4 Performance Estimation Model

To predict the performance behavior of the hybrid indexing tree, we have developed a performance estimation model. Our model is divided into two parts: the first part for estimating the key parameters of the tree (e.g., the characteristics of HMBRs of tree

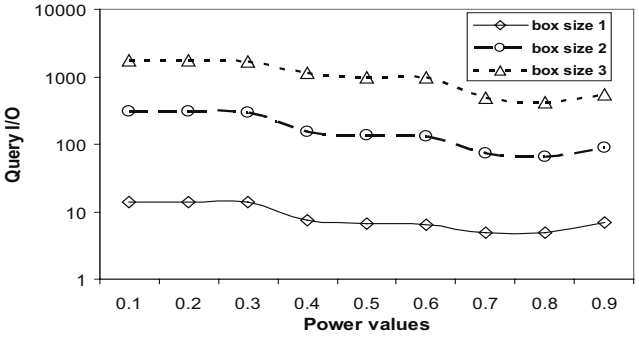


Fig. 5. Performance for enhanced strategy with power value adjustment

nodes) for a given HDS; and the second part for estimating the number of I/Os for arbitrary query box sizes based on the key parameters. If the tree is given and we want to predict the performance behavior of the tree, only the second part is needed. The two parts of our model are sketched as follows.

Part I: Estimating key parameters of a tree

In this part of the performance model, we are given an HDS and the number of vectors (to be indexed) in the HDS. Hence we have the following input parameters: the system disk block size, the number of continuous and discrete dimensions, the domain/alphabet size of the discrete domains, and the number of vectors to be indexed. We assume that the components of the vectors are uniformly distributed in their respective dimensions.

Using the above information, we can first determine the maximum number of entries M_n in a non-leaf node, and the maximum number of entries M_l in a leaf node. From the heuristics extended to the HDS, we notice that the numbers of splits different nodes (at the same level of the tree) went through usually either equal to each other or differ by 1. The tree growing process involves two time windows: during any time of the tree growth, when a node at certain level splits, all the other nodes at the same level split around the same time. We call this time period *the splitting (time) window*. After a node is split, the new nodes start to accumulate incoming data for some time. We call this period *the accumulating (time) window*. Since a new node created from a split is about half-full and takes quite some time before it becomes full again, the accumulating window is typically much larger than the splitting window. Thus we focus on capturing the performance behavior for the accumulating window in our performance model.

At the beginning of an accumulating window the node space utilization is about 50% because each overflow node has split into two half-full nodes. When the accumulating window ends, the node space utilization is close to 100%. On average the utilization would be 75%. However, we notice that, in real world situations, although most splits occur in the splitting window, there are some splits happening during the accumulating window. As a result, the number of nodes during the accumulating window is slightly more than what is estimated under the assumption that all splits happen in the splitting window. Hence the actual average node space utilization (around 70% from our experiments) is also lower than expected. Therefore, our estimates for the average numbers

of entries in a non-leaf node and a leaf node during the accumulating window are: E_n = 0.7 * M_n and E_l = 0.7 * M_l, respectively.

The height h of the tree to index V number of vectors is estimated as: $h = \lceil log(\lceil V/E_l \rceil)/\lfloor logE_n \rfloor \rceil + 1$. The estimated number of nodes at the leaf level is $n_0 = \lceil V/E_l \rceil$. The estimated number of nodes at level i is: $n_i = \lceil n_{i-1}/E_n \rceil$ $(1 \leq i \leq h)$. The number of rounds of splits that every node at level i have gone through can be estimated as: $w_i = \lfloor log_2(n_i) \rfloor$. The number of nodes at level i which have gone through one more round of split is estimated as: $v_i = (n_i - 2^{w_i}) \times 2$.

After we know the height of a tree, the number of nodes at each level and the number of splits each node has gone through, we can estimate the parameters (e.g., edge length) for the HMBRs of nodes at each level. Details of the lengthy derivation are omitted here due to the space limitation.

Part II: Estimating query performance based on key parameters of a tree

The second part of our model estimates the number of I/Os needed for a hybrid indexing tree (defined by the key parameters discussed in *Part I*) given arbitrary query box sizes. The estimation has three steps:

(ES-1) Estimating the overlapping probability for one discrete/continuous dimension

For a discrete dimension d, assume that the domain size of the dimension is D_d, the set size of a node $N's$ HMBR on dimension d is L_d, and the set size of a query box on this dimension is T_d. Clearly, $L_d \leq D_d$ and $T_d \leq D_d$. The probability of the query box overlapping with $N's$ HMBR on dimension d is:

$$1 - C_{D_d - L_d}^{T_d} / C_{D_d}^{T_d}$$

For a continuous dimension c, without loss of generality, assume that the domain range/interval is $[0, C_c]$, and the lower and upper bounds of a node $N's$ HMBR on dimension c are L_c and U_c, respectively. Further suppose the edge length of a query box on this dimension is T_c. We have $0 \leq L_c, U_c \leq C_c$ and $T_c \leq C_c$. The probability for the query box to overlap with $N's$ HMBR on dimension c is:

$$(b - a)/(C_c - T_c)$$

where $a = max\{L_c - T_c,\ 0\}$ and $b = min\{U_c,\ C_c - T_c\}$. This probability calculation is based on the lower bound value p of the query box on dimension c. Clearly, p is within range $[0,\ C_c - T_c]$. If the query box has an overlap with interval $[L_c, U_c]$ on dimension c, p must be within $[L_c - T_c,\ U_c]$. a and b are used to handle boundary conditions when $(L_c - T_c) < 0$ and $(U_c + T_c) > C_c$.

(ES-2) Estimating the overlapping probability for one tree node

The probability of a tree node N overlapping with an arbitrary query box Q, is the product of the overlapping probabilities of N and Q on all dimensions, which are calculated by *ES-1*.

(ES-3)Estimating the I/O number for the tree

The number of I/Os for the tree to process a box query is estimated as the summation of the overlapping probabilities between the query box and every tree node, which could be calculated by *ES-2*.

We conducted experiments to verify the above performance model. Two sets of typical experimental results are shown in Figures 6 and 7. Each observed performance data

was measured using the average number of I/Os for 200 random queries. The HDS used in the experiments has 4 continuous dimensions and 4 discrete dimensions with an alphabet size of 10.

Figure 6 shows the comparison between our estimated and observed I/O numbers for queries with box sizes $1 \sim 3$. The number of indexed vectors ranges from 1 million to 10 million. Since the trees are given, the experimental results actually demonstrate the accuracy of the second part of our performance model. The experimental results show an average relative error of only 2.45% in such a case.

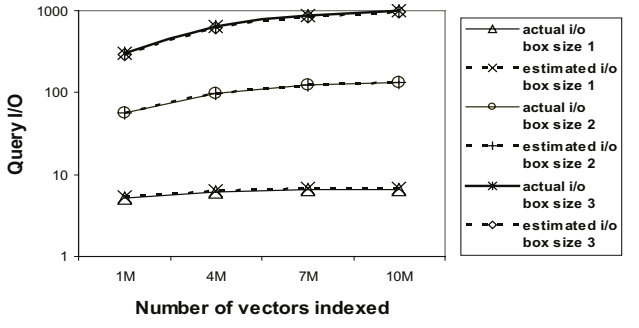


Fig. 6. Verification of performance model for given trees and given HDSs

Figure 7 shows the comparison between our observed I/O numbers (from queries on actual trees) and estimated I/O numbers (for the given HDS without building any tree). In this case, the tree parameters for the given HDS also need to be estimated using the first part of our model. From the figure, we can see that our performance model still give quite good estimates although the accuracy degrades a little bit due to the fact that more parameters need to be estimated. The average relative error is 5.76%.

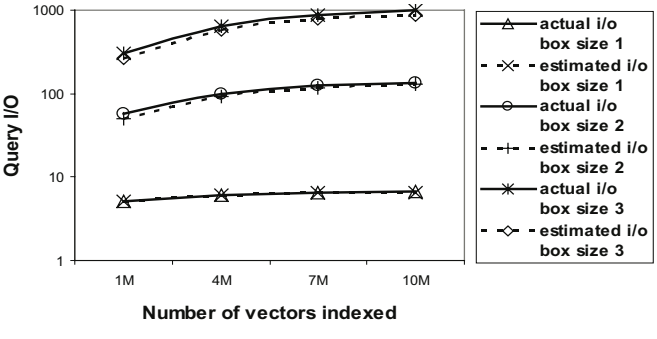


Fig. 7. Verification of performance model for given HDSs

5 Conclusions

In this paper, the original ND-tree structure and its building heuristics are extended to the HDS. A power value adjustment strategy is employed to make the measures on

continuous and discrete dimensions comparable and controllable. A theoretical model is also developed to predict the performance of the hybrid indexing in HDSs.

Our experimental results demonstrate that the extended ND-tree's heuristics are still effective in supporting box queries in the HDS. Using these heuristics to index the HDS is more efficient than the traditional linear scan, the method to index the continuous subspace of the underlying HDS using the R*-tree and the method to index the discrete subspace using the ND-tree. The reason is that during the query time the hybrid indexing approach could prune nodes based on information from additional dimensions which the R*-tree and ND-tree do not have.

Our future work includes developing more effective heuristics for the HDS indexing.

Acknowledgements. This research was supported by the US National Science Foundation (under grants # IIS-0414576 and # IIS-0414594), Michigan State University and the University of Michigan.

References

1. Beckmann, N., Kriegel, H.-P., Schneider, R., Seeger, B.: The R*-tree: an efficient and robust access method for points and rectangles. In: Proceedings of ACM SIGMOD, pp. 322–331 (1990)
2. Catlett, J.: On changing continuous attributes into ordered discrete attributes. In: Proceedings of the European Working Session on Machine Learning, pp. 164–178 (1991)
3. Chakrabarti, K., Mehrotra, S.: The hybrid tree: an index structure for high dimensional feature spaces. In: Proceedings of the 15th International Conference on Data Engineering, pp. 440–447 (1999)
4. Chen, C., Pramanik, S., Watve, A., Zhu, Q., Qiang, G.: The C-ND Tree: A Multidimensional Index for Hybrid Continuous and Non-ordered Discrete Data Spaces. In: Proceedings of the 12th International Conference on Extending Database Technology (2009)
5. Freitas, A.A.: A survey of evolutionary algorithms for data mining and knowledge discovery. In: Advances in Evolutionary Computing: Theory and Applications, pp. 819–845 (2003)
6. Guttman, A.: R-trees: a dynamic index structure for spatial searching. In: Proceedings of ACM SIGMOD, pp. 47–57 (1984)
7. Henrich, A.: The LSDh-tree: an access structure for feature vectors. In: Proceedings of the 14th International Conference on Data Engineering, pp. 362–369 (1998)
8. Macskassy, S.A., Hirsh, H., Banerjee, A., Dayanik, A.A.: Converting numerical classification into text classification. Artificial Intelligence 143(1), 51–77 (2003)
9. Qian, G., Zhu, Q., Xue, Q., Pramanik, S.: The ND-tree: a dynamic indexing technique for multidimensional non-ordered discrete data spaces. In: Proceedings of the 29th International Conference on VLDB, pp. 620–631 (2003)
10. Qian, G., Zhu, Q., Xue, Q., Pramanik, S.: Dynamic indexing for multidimensional non-ordered discrete data spaces using a data-partitioning approach. Proceedings of ACM Transactions on Database Systems 31(2), 439–484 (2006)
11. Qian, G., Zhu, Q., Xue, Q., Pramanik, S.: A space-partitioning-based indexing method for multidimensional non-ordered discrete data spaces. ACM Trans. on Information Syst. 23(1), 79–110 (2006)
12. Robinson, J.T.: The K-D-B-tree: a search structure for large multidimensional dynamic indexes. In: Proceedings of ACM SIGMOD, pp. 10–18 (1981)

Relaxing XML Preference Queries for Cooperative Retrieval

SungRan Cho and Wolf-Tilo Balke

L3S Research Center, Leibniz University of Hannover, 30167 Hannover, Germany
{scho,balke}@L3S.de

Abstract. Today XML is an essential technology for knowledge management within enterprises and dissemination of data over the Web. Therefore the efficient evaluation of XML queries has been thoroughly researched. But given the ever growing amount of information available in different sources, also querying becomes more complex. In contrast to simple exact match retrieval, approximate matches become far more appropriate over collections of complex XML documents. Only recently approximate XML query processing has been proposed where structure and value are subject to necessary relaxations. All the possible query relaxations determined by the user's preferences are generated in a way that predicates are progressively relaxed until a suitable set of best possible results is retrieved. In this paper we present a novel framework for developing preference relaxations to the query permitting additional flexibility in order to fulfil a user's wishes. We also design IPX, an interface for XML preference query processing, that enables users to express and formulate complex user preferences, and provides a first solution for the aspects of XML preference query processing that allow preference querying and returning ranked answers.

Keywords: XML query processing, Preference-based retrieval, Personalization.

1 Introduction

XML is widely used as a base technology for knowledge management within enterprises and dissemination of data on the Web (like e.g., product catalogues), because it allows to organize and handle semistructured data. A collection of XML documents is viewed as a forest of node labeled trees. The data generally can be queried on both structure and content using advanced retrieval languages such as XPath or XQuery. User queries will usually be structured to express the user's information needs and users often have quite specific preferences about the structure, especially when the document structure shows a certain semantics often described by DTDs or XML schema. However, due to the large number and complexity (or heterogeneity) of XML documents, the retrieval process should be cooperative between system and user. Here approximate matches that allow to rank answers according to their relevance to the query [3, 6, 19, 20] are more appropriate than exact match queries. Recently several proposals have therfore studied ranking methods that account for structure to score answers to XML queries.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 160–171, 2009.
© Springer-Verlag Berlin Heidelberg 2009

When querying for information users may only have a vague idea of what type of documents can be expected in the collection as well as where the information might occur in a specific XML document. Personal preferences are a powerful means to express user wishes that might not always be fulfilled, but allow for relaxation of query predicates by user provided alternatives to search desired information and order the search results. Hence, preferences combat two undesirable scenarios: to return an empty result and to flood the user with too many results. Relaxation of overspecified queries avoids empty results and to cope with flooding a pruning of less relevant answers can be performed (top-k querying).

In this paper we focus on the *relaxation of structural preferences* in a query, inspired by fair query relaxation techniques applied to the query structure [2]. In order to avoid empty results and to further personalize user queries, a preference query considers node relaxations to the preferred query structure to return 'closest' (or the most relevant) results to the user request. Preference queries are rewritten into a set of queries progressively posed to the database (unfolding): starting with a highly specific query with all top attributes from a user's preferences, each predicate is gradually relaxed to less preferred attributes (*base relaxation*). Moreover, preferred structures in the query can be relaxed to all still relevant query structure (*secondary relaxation*).

Since preferences generally induce a certain ranking on the result set, an unfolding sequence can be enumerated by successively relaxing query predicates from most to least preferred attributes in each preference, considering all variations as query rewritings, and collecting the results with respective rank information. In this paper we additionally define an ordering scheme that provides a fair relaxation by taking into account the order induced by not only multiple preference attributes (structure and value), but also their relaxed versions. Efficient preference XML query processing now requires the ability to execute a set of relevant queries and determine most relevant results as induced by user provided preferences. We design IPX, an interface system for effective XML preference query processing. IPX incorporates XML query processing enhanced with preference features providing not only a extension of XPath syntax, but also dynamic preference operations such as an ordering method and top-k processing.

2 XML Preference Queries

For a complete overview of our structure-based XML preference framework see [7]. We consider a data model where information is represented as a forest of node labeled ordered trees. Each non-leaf node in the tree has a type as its label, where types are organized in a simple inheritance hierarchy. Each leaf node has a string value as its label. Simple data instances are given in Figure 1.

Beyond simple lookups modern database retrieval enables users to express certain preferences. Such preferences state what a user likes/dislikes and preference query processing attempts to retrieve all best possible matches in a cooperative fashion avoiding empty result sets. In relational databases such preferences are specified over data values. But, since in XML retrieval also the structure of the document often carries semantics described by a DTD or XML schema, user preferences can also be specified over the structure. Qualitative preferences are often expressed by partial

orders and visualized by a graph where each node is labeled by values of a particular type and the direction of each edge between nodes expresses that the label of the node where the edge originates, is preferred over the label to which the edge points. The example structural preference of Figure 2(b), SP_{treat}, for instance, expresses that *toy* is preferred over *hair-care*.

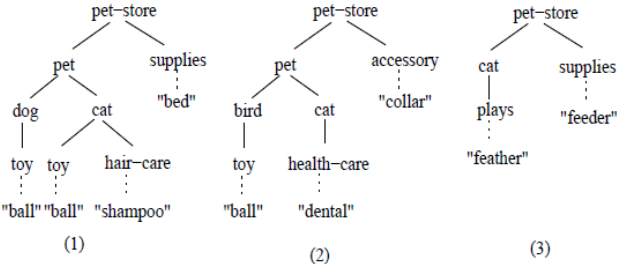


Fig. 1. Example for heterogeneous XML databases

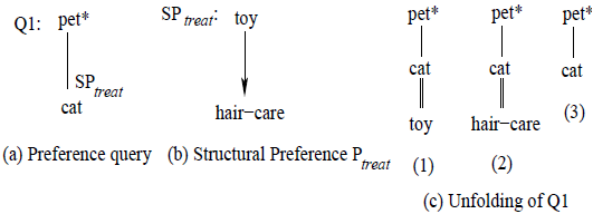


Fig. 2. Example query and structural preference

Like XML documents, queries are usually structured to express the users' information needs. All existing query languages for XML are tree-shaped, where nodes are labeled by types or string values, and edges correspond to parent-child or ancestor-descendant relationships. Figure 2 (a) shows a simple path query with a preference: each edge represents a parent-child relationship (a double edge is an ancestor-descendant relationship). Some nodes are marked with '*', indicating that they are projected (i.e. returned as answer). Any leaf nodes can be marked with explicit preferences.

A preference query then needs to be rewritten as an ordered set of queries induced by the explicit preference (referred to as query process), retrieving all data relevant to answering the original query. We thus have to examine all possible relaxations of the user preferences and expand the preference-marked nodes accordingly. Next we define the unfolding of preference queries for query processing in case of a single preference in a query:

Definition 2.1. [*Unfolding Preference Queries*] Given a query Q with a node n marked with a preference P, an expanded query Q' is obtained by unfolding Q:

(i) while respecting the order induced by P adding a node v of P to a query node n as a successor (in the case where n is value-constrained, v resides between n and the

value), adding an edge with ancestor-descendent label from n to v, and propagating a possible distinguished status of n down to v,
(ii) entirely removing P from Q.

This results in an ordered set of $|P + 1|$ distinct queries: the most preferred query is expanded by (one of) the most preferred node(s) in the preference graph and the least preferred query of the expansions is obtained by simply removing mark P from query node n without adding any new nodes and edges. An incoming edge of a preference node in a query is generalized to an ancestor-descendant relationship allowing all matchings to relevant portions of the XML database.

Given query Q_1 in Figure 2(a) containing a structural preference SP_{treat}, the induced order providing a desirable stepwise relaxation scheme is reflected by all possible expansions of Q_1 in Figure 2 (1), (2), and (3) in the unfolding process. Starting with query (1), whose answer set the user prefers most over queries (2) and (3), and down to query (3) as the least preferred query. With only a single preference considered, we encounter 3 different rewritings for the query in the unfolding process.

Generally speaking, preference attributes whether they are value or structural information, are progressively relaxed, i.e., starting with all top attributes stated in a user's preference and gradually relaxing to less preferred attributes. To capture preferred answers to a given preference query we first generate the most specific relaxed preference queries on a set of queries:

Definition 2.2. [*Base Relaxation*] Let Q be an XML preference query. Then query Q' expanded with preference nodes from P in the unfolding process is referred to as base relaxation. ■

For example, given the query 2, the queries generated from the base relaxations are Figure 2(c) (1) and (2) (but not (3)). Our focus is closely related to the issue of loss of desirability, i.e. how much a preference query is relaxed. To address this problem we next propose a framework for relaxing structural preferences in an organized way.

3 Relaxing Queries Based on Structural Patterns

In this section we formally define approximate preferred queries based on the notion of structural pattern relaxations. As a base, we take the node generalization relaxation defined in [2], and generalize it in the context of preferences. In particular, we consider three specific preference relaxations that all are structural relaxations of the preference query. A set of predefined base preferences is provided by each user, all of which can be described as partial order graphs and alternative sets of preferences are generated with respect to a notion of structural similarity. We consider relaxing preference nodes in the query, which we refer to as secondary relaxation. Such secondary relaxation is closely related with similarity searches. While usually an ontology plays a central role in a similarity search, here we simply use DTD or XML Schema information to promote the relevant relaxed nodes.

Node Generalization. *This permits the type of each preference node to be generalized to a supertype. For example consider the expansion query of Figure 2(c) (1) and a type hierarchy containing the path "toy|plays|necessity". The toy can be generalized to*

plays followed by necessity, allowing for arbitrary play and pet necessities to be returned instead of just toy information. This kind of node relaxation is appropriate, whenever no exact match is found.

Sibling Promotion. *This permits the corresponding sibling DTD nodes of each preference node to be promoted. For example, if the query of Figure 2(c) (1) does not result in a match, but there may be a lot of cat related products that come very close to a toy. Thus, this near preferred sibling node could be relevant, if no exact match is available.*

Path Node Promotion. *This also uses DTD information to promote relevant relaxed nodes. It permits the nodes to be promoted on the corresponding DTD path between a query node and an expanded preference node (excluding the original query and preference nodes themselves). For example, consider the query in Figure 2(c) (1). If there is a node product on the DTD path between the corresponding cat and toy, in the near node product can be promoted as an alternative closest node of the toy node.*

Preference queries have to be relaxed into a set of queries that is guaranteed to retrieve every possibly relevant document from an XML database. In the relaxations above, only preference nodes in the query are relaxed and approximate matches in the corresponding relaxed nodes are retrieved. These relaxations do not increase the query size, but increase the set of those candidate queries close to the original query.

So far, we have focused on generalizations of preference nodes while keeping a generalized edge of ancestor-descendant type. However, users might wish to retrieve preferred answers associated with some edge information. In order to permit an additional specification in preferences, we allow users to specify if edges should be constrained with respect to their depth.

Edge Depth Preference. *This permits a depth range of the appended preference node in order to limit the search radius. For example, in the query 2(c) (1), the user could constrain the edge (cat, toy) with maximum depth information, retrieving only toys that are within a certain distance from the cats.*

4 Ordering Relaxations

An ordering method is necessary to distinguish different relaxations of the initial user query and get a notion of what is a *minimum amount of relaxation*. For ranking, several techniques have been proposed, like e.g., approximate keyword queries based on ontologies [22], or the *tf*idf* measure of the IR community that matches keyword queries against a document collection. In any case, the general approach to relaxation is in line with our development in the paper.

Our base relaxation ordering method uses Pareto optimality to rank all combinations of preference nodes and their relaxed versions expanded in the query. But the challenge in this paper is also to organize a framework for ordering a total relaxation of a preference query, i.e. including secondary relaxations. We will now define an explicit relaxation order that distinguishes the loosened preference nodes.

Definition 4.1. [*Relaxation Order*] Let Q be an XML preference query and Q' be an expansion query. Then the preference node is relaxed in the following sequence:

1. *node generalization*: replaces its immediate supertype of the preference node recursively;
2. *sibling promotion*: replace its corresponding DTD sibling nodes of the preference node;
3. *path node promotion*: replaces the parent of the preference node in the corresponding DTD path recursively;
4. *preference node deletion*: finally deletes the preference node. ■

While Definition 4.1 (2) takes into account the local closeness that treats all siblings of a given preference node equally, Definition 4.1 (1), (3), and (4) account for the subsumed closeness of the corresponding preference node. Intuitively, node generalization is considered as the most specific (or most precise) relaxation, since subsuming nodes are very closely related to the preferred node. The sibling promotion is next, because it leads to a less specific, but often still relevant relaxation, and the DTD path node promotion is last, because it is a generalization of sibling promotion.

The sequence of our relaxation can quantify the closeness of an answer in the collection of documents. In our framework, the ranking method is monotonic since each relaxation step always follows the dominance relation and the relaxed query includes all the nodes from the original query, i.e., only relaxing expanded nodes. For example, given the query Q_1 in Figure 2(a), Figure 3 shows the sequence of different relaxation to the preference nodes *toy* and *hair-care*.

In our basic relaxation ordering scheme, we still need to distinguish the degree of subsumed closeness in order to increase precision of ranking. The general rationale for this closeness is to use the distance on the DTD path or the path in the type hierarchy between the original query node and the relaxed preference node, i.e., if the distance gets larger, the degree of closeness to the initial query decreases. The distance

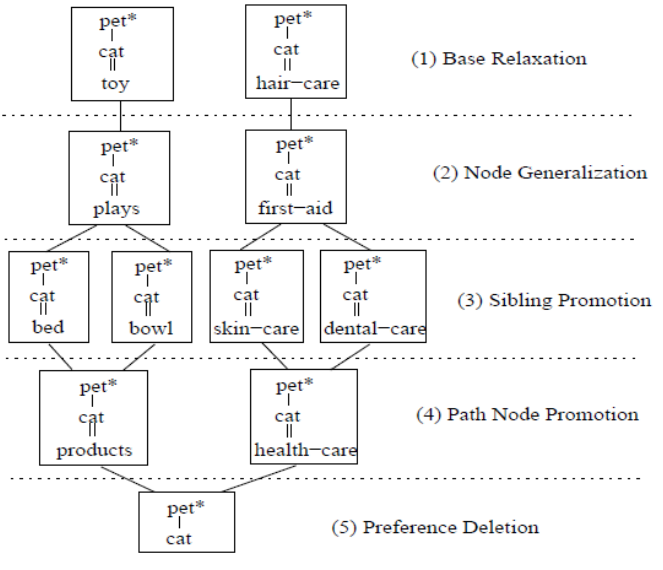


Fig. 3. Example preference relaxation process

measure becomes useful to compute the score (or rank) of answers to decide how closely they match the query. If a query node contains a single preference P, where P consists of $|P|$ distinct attributes, the total number of relaxed queries from the secondary relaxation is:

$$\sum_{i=1}^{|P|} |\, Sibling_i \,| + |\, SuperType_i \,| + |\, DTDpathNode_i \,|$$

The generalization of definition 4.1 to the case of multiple nodes marked with preferences is straightforward: each single preferred node is relaxed such that every possible combination generated from base and secondary relaxations is reflected by a relaxed query.

5 Dealing with Multiple Preferences

In this section, we discuss incorporating multiple preferences, especially in the case where a query node is marked with multiple preferences. The unfolding process of the query should be based on the semantics of the query we consider. This is a generalization of definition 2.1. If query nodes are marked with a single preference, the total number of queries relaxed from the base relaxation is $|P_1+1| \times \ldots \times |P_n+1|$, where each preference P_i consists of $|P_i|$ attributes.

Individual users may have specific structural and value preferences. For evaluation it is necessary to combine different preferences specified on a single query node. To explain the basic idea, we consider the case where query nodes are marked with a single structure preference and a single value preference together in this subsection.

If a query node is marked with a single value preference, an unfolding of the query is the same as we have shown in Figure 2. However, in the case where a query node can be marked with structural and value preferences together, we need to expand the query with a set of queries in a way to comply with the semantics of query described in Section 2. Thus, structural elements should be expanded prior to values because in XML data and queries only leaf nodes specify their values.

For example, consider the query Q_2 in Figure 4(a), where Q_2 contains the structural preference SP_{treat} in Figure 2(a) and the value preference VP_{toy} in Figure 4(b). The query Q_2 is rewritten into a set of queries in Figure 4(c) by expanding structural elements followed by values. The query Q_2 produces 9 possible expansions in Figure 4(1)-(9) in the unfolding process.

Next we discuss additional flexibility of structural preferences in the query. Multiple preferences can be specified on a single query node as well as on multiple query nodes in the query. For example, consider a query consisting of *cat* marked with two structural preferences (for ease of understanding we will use two times SP_{treat}). We now need to consider all possible query structures encountered. Figure 5 shows the relaxed queries. A key part of defining a relaxation order is to examine all possible structure combinations in the query, which is shown in the query in Figures 5(1)-(4). In particular the queries in Figures 5(3) and (4) are valid, especially when the order of queries is material. However if the order of queries is not concerned, Figures 5(3) and (4) are equivalent because they return the same answers. However by referencing the

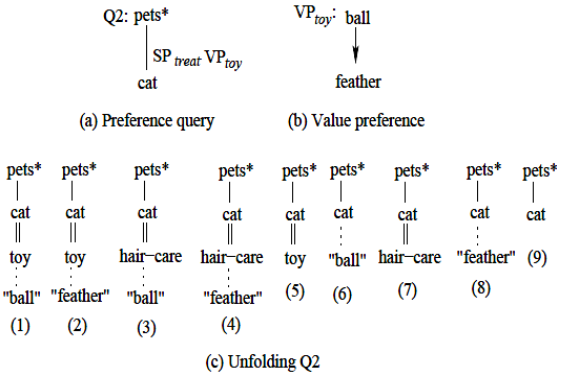


Fig. 4. Example query with structural and value preferences

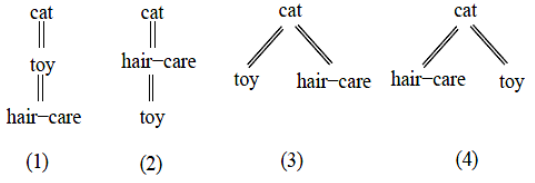


Fig. 5. Example of multiple structural preferences

DTD, the number of relaxed queries encountered in the base and secondary relaxation processes may be reduced. For preferences on element tags, the respective DTD can help to prune a set of relaxed queries by simply testing if the relaxed node is valid in the query structure.

6 Design of IPX

For evaluating our concepts, we designed an interface called IPX to enhance queries with preference operations. Since preferences are specified on top of query expressions in languages such as XPath and XQuery, IPX can be implemented on top of commercial XML servers supporting such queries. Figure 6 shows the overall architecture of IPX. IPX is mainly composed of three components: query rewriter, preference handler, and ranking handler. The IPX architecture has successfully been demonstrated at ACM SAC 2009, see [8].

The IPX first accepts a user query containing structural and/or value preferences. Then the *query rewriter* rewrites the query into conventional XPath or XQuery queries which can be executed in any conventional XML engine, by expanding the query with the user provided preferences. In particular, if a query contains structural preferences, the rewriter checks the corresponding DTD to expand relevant elements. Since IPX handles ordered answers, the *ranking handler* determines the necessary set of queries and preserves the induced order of the result set by quantifying the relevance

of the relaxed queries. In addition, since structural and value preferences of individual users need to be maintained, the *preference handler* parses them and stores them in a repository. It also is designed to manage multiple granularities of preferences to interact with multiple preference functions given in the query, and to compute their conjunction currently following the Pareto semantics (of course our framework is open for extensions). Moreover, it interacts with the query rewriter to support the unfolding process. Furthermore IPX encompasses the following functionality and features:

Extending XPath syntax: IPX implements several flexible syntaxes, which incorporate preference assignment to the query and necessary preference operations.

Supporting top-k processing: IPX also supports the efficient evaluation of top-k queries that retrieve only the k best answers. The *preference handler* and *ranking handler* allow to synchronize the top-k retrieval.

Query optimizer: Since a preference has to be unfolded into a set of queries and such expansions typically contain redundancies, it is important to identify and simplify necessary relaxed queries for effective evaluation. IPX implements a *preference query optimizer* that not only determines an optimal set of expansion queries, but also preserves an ordering induced by the preference.

Improving query evaluation: IPX implements evaluation techniques to improve evaluation times for a preference query by considering the special features of preference queries that typically contain repetitive fragments and always follow induced patterns.

Visualization: IPX implements a flexible and interactive graphical interface that facilitates browsing of different preference graphs and DTDs, and displays user queries as well as the results of query evaluations. For example, given the query shown in the left window of Figure 7, some queries in the unfolding process are displayed in the right window of Figure 7, where the first relaxed query is the most preferred query, the second and the third queries are next and etc.

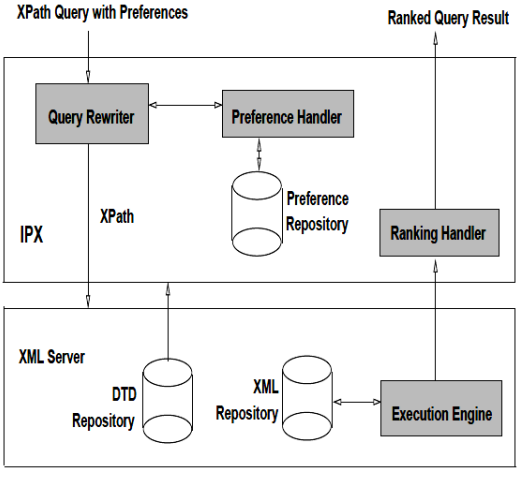


Fig. 6. Overall architecture

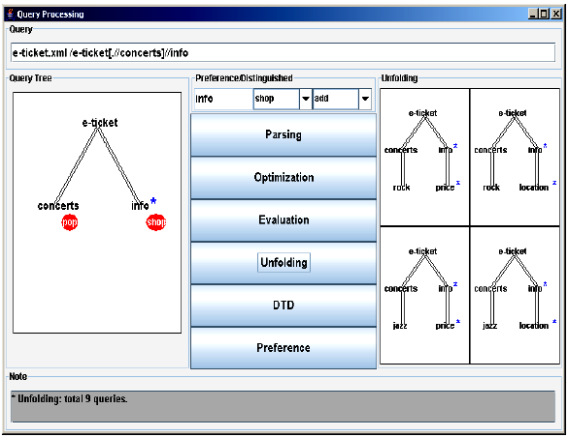


Fig. 7. IPX Preference XML query processor

7 Related Work

Preferences recently are an active research area in information systems research [9, 1, 14, 15, 21, 13]. Due to the importance in practical applications, preference query processing has attracted considerable attention leading to a large number of possible techniques.

Several systems for supporting preference queries have been recently proposed. In [10], the author investigated the semantic optimization of preference queries to remove redundant occurrences of preferred values in relational databases. Recent work by [15] proposed preference XML queries in connection with the modeling framework for preferences as partial order graphs given in [14]. The resulting language enables the use of soft filtering conditions in contrast to conventional exact match conditions for node-selection in XPath. Like in our approach a soft condition defines a strict partial order over the set of elements to be filtered and then returns only the best matches. In [4] authors proposed a fair scheme of relaxing preference values to pose to the database such that the user retrieves data in a well-defined order and can choose at any stage, whether he is already satisfied by the result and the query processing can be terminated. Another interesting study is presentational preferences for XML query results. Although in contrast to our approach these preferences do not affect the matching process, some basic techniques exploiting the DTD structure are related. Since XQuery is a data-transformation query language, users can easily define a set of preferences to impose an ordering on the presentation of the results.

Another related area of work is scoring. In particular scoring for XML databases has actively been studied [2, 5, 6, 19, 20]. While our approach gives a fair relaxation method, these approaches promote traditional scoring methods for XML such as probability-based approaches, considering path expressions along with query keywords, and incorporating a similarity measure. Recognizing an exact match retrieval model as often inadequate in practical scenarios, the resulting XML query engines (e.g., XIRQL [11] and XXL [22]) have already been extended to support an IR-style

matching of data values. Generally these engines allow finding similar values for a certain predicate or relaxing query predicates to find desired values in related structural elements.

There are also several XML query relaxation proposals (see, e.g., [2, 3]). In particular, [2] addressed the problem of approximate XML query matching based on tree pattern query relaxations and provided efficient algorithms to prune query answers that will never meet a given threshold. While before the focus was on defining a framework for structural relaxation, the work in [3] focuses on scoring methods on both structure and content to evaluate top-k answers to XML queries in the same relaxation framework. Building on previous approaches for preferences in information systems, in this paper we presented a framework for how XML preference queries can be relaxed not only by such preference information, but also by similarity of given preferred information.

8 Conclusions

In this paper we presented a framework for relaxing and combining structural preferences to search personalized information on XML databases or semi-structured document collections like e.g., enterprise document collections or e-catalogues.

In order to fulfill the user's interest to the best possible degree, we considered preference query relaxations to retrieve closest relevant results to a user request. We showed that our framework can be applied to existing XPath engines by a syntactic enhancement that incorporates function calls in the query. We also designed and implemented IPX, a flexible interface for handling preferences in the query. IPX is extensible to incorporate other necessary applications such as automatic extractions of structural preferences using DTDs and/or user profiles. IPX thus can help sharing efforts when developing preference XML engines.

We are currently investigating extensions to IPX to support conventional scoring methods which account for both structure and value. At the same time we also address the problem of how to score answers for complex joint matches. In order to improve efficiency of the overall preference query processing, we see many interesting directions of future work such as efficient preference query evaluation and optimization strategies. Solutions to these problems will contribute to the application of preference-based personalization in XML queries.

References

1. Agrawal, R., Wimmers, E.: A framework for ex-pressing and combining preferences. In: ACM SIGMOD Conference on Management of Data, Dallas, TX, USA (2000)
2. Amer-Yahia, S., Cho, S., Srivastava, D.: Tree Pattern Relaxation. In: Jensen, C.S., Jeffery, K., Pokorný, J., Šaltenis, S., Bertino, E., Böhm, K., Jarke, M. (eds.) EDBT 2002. LNCS, vol. 2287, p. 496. Springer, Heidelberg (2002)
3. Amer-Yahia, S., Koudas, N., Marian, A., Srivastava, D., Toman, D.: Structure and content scoring for XML. In: Int. Conference on Very Large Databases (VLDB), Trondheim, Norway (2005)

4. Balke, W., Wagner, M.: Through different eyes: assessing multiple conceptual views for querying Web services. In: World Wide Web Conference (WWW), New York, NY, USA (2004)
5. Bremer, J., Gertz, M.: XQuery/IR: Integrating XML document and data retrieval. In: WebDB, Madison, WI, USA (2002)
6. Chinenyanga, T., Kushmerick, N.: Expressive and efficient ranked querying of XML data. In: WebDB, Santa Barbara, CA, USA (2001)
7. Cho, S., Balke, W.: Order-Preserving Optimization of Twig Queries with Structural Preferences. In: IDEAS, Coimbra, Portugal (2008)
8. Cho, S., Balke, W.: Building an Efficient Preference XML Query Processor. In: ACM Symposium on Applied Computing (SAC), Honolulu, HI, USA (2009)
9. Chomicki, J.: Querying with intrinsic preferences. In: Jensen, C.S., Jeffery, K., Pokorný, J., Šaltenis, S., Bertino, E., Böhm, K., Jarke, M. (eds.) EDBT 2002. LNCS, vol. 2287, p. 34. Springer, Heidelberg (2002)
10. Chomicki, J.: Semantic optimization of preference queries. In: Int. Symposium on Applications of Constraint Databases, Paris, France (2004)
11. Fuhr, N., Großjohann, K.: XIRQL: A query language for information retrieval in XML Documents. In: ACM SIGIR, New Orleans, LA, USA (2001)
12. Guo, L., Shao, F., Botev, C., Shanmugasundaram, J.: XRANK: Ranked keyword search over XML document. In: ACM SIGMOD Conference on Management of Data, San Diego, CA, USA (2003)
13. Kanza, Y., Sagiv, Y.: Flexible queries over semi-structured Data. In: Int. Symposium on Principles of Database Systems (PODS), Santa Barbara, CA, USA (2001)
14. Kießling, W.: Foundations of preferences in database systems. In: Int. Conference on Very Large Databases (VLDB), Hong Kong, China (1999)
15. Kießling, W., Hafenrichter, B., Fischer, S., Holland, S.: Preference XPATH: a query language for E-commerce. In: Konferenz für Wirtschaftsinformatik Augsburg, Germany (2001)
16. Koch, C., Scherzinger, S., Schweikardt, N., Stegmaier, B.: FluXQuery: an optimizing XQuery processor for streaming XML data. In: Int. Conference on Very Large Databases (VLDB), Toronto, Canada (2004)
17. Papadias, D., Tao, Y., Fu, G., Seeger, B.: An optimal and progressive algorithm for skyline queries. In: ACM SIGMOD Conference on Management of Data, San Diego, CA, USA (2003)
18. Papakonstantinou, Y., Vassalos, V.: Query rewriting for semi-structured data. In: ACM SIGMOD Conference on Management of Data, Philadelphia, PA, USA (1999)
19. Polyzotis, N., Garofalakis, M., Ioannidis, Y.: Approximate XML Query Answers. In: ACM SIGMOD Conference on Management of Data, Paris, France (2004)
20. Schlieder, T.: Schema-driven evaluation of approximate tree-pattern queries. In: Jensen, C.S., Jeffery, K., Pokorný, J., Šaltenis, S., Bertino, E., Böhm, K., Jarke, M. (eds.) EDBT 2002. LNCS, vol. 2287, p. 514. Springer, Heidelberg (2002)
21. Stolze, M., Rjaibi, W.: Towards scalable scoring for preference-based item recommendation. In: Bulletin of the IEEE Technical Committee on Data Engineering (2001)
22. Theobald, A., Weikum, G.: The index-based XXL search engine for querying XML Data with relevance ranking. In: Jensen, C.S., Jeffery, K., Pokorný, J., Šaltenis, S., Bertino, E., Böhm, K., Jarke, M. (eds.) EDBT 2002. LNCS, vol. 2287, p. 477. Springer, Heidelberg (2002)

DeXIN: An Extensible Framework for Distributed XQuery over Heterogeneous Data Sources*

Muhammad Intizar Ali[1], Reinhard Pichler[1],
Hong-Linh Truong[2], and Schahram Dustdar[2]

[1] Database and Artificial Intelligence Group, Vienna University of Technology
{intizar,pichler}@dbai.tuwien.ac.at
[2] Distributed Systems Group, Vienna University of Technology
{truong,dustdar}@infosys.tuwien.ac.at

Abstract. In the Web environment, rich, diverse sources of heterogeneous and distributed data are ubiquitous. In fact, even the information characterizing a single entity - like, for example, the information related to a Web service - is normally scattered over various data sources using various languages such as XML, RDF, and OWL. Hence, there is a strong need for Web applications to handle queries over heterogeneous, autonomous, and distributed data sources. However, existing techniques do not provide sufficient support for this task. In this paper we present DeXIN, an extensible framework for providing integrated access over heterogeneous, autonomous, and distributed web data sources, which can be utilized for data integration in modern Web applications and Service Oriented Architecture. DeXIN extends the XQuery language by supporting SPARQL queries inside XQuery, thus facilitating the query of data modeled in XML, RDF, and OWL. DeXIN facilitates data integration in a distributed Web and Service Oriented environment by avoiding the transfer of large amounts of data to a central server for centralized data integration and exonerates the transformation of huge amount of data into a common format for integrated access.

Keywords: Data integration, Distributed query processing, Web data sources, Heterogeneous data sources.

1 Introduction

In recent years, there has been an enormous boost in Semantic Web technologies and Web services. Web applications thus have to deal with huge amounts of data which are normally scattered over various data sources using various languages. Hence, these applications are facing two major challenges, namely (i) how to integrate *heterogeneous* data and (ii) how to deal with *rapidly growing* and continuously changing *distributed data sources*.

The most important languages for specifying data on the Web are, on the one hand, the Extensible Markup Language (XML) [1] and, on the other hand, the Resource Description Framework (RDF) [2] and Ontology Web Language (OWL) [3]. XML

* This work was supported by the Vienna Science and Technology Fund (WWTF), project ICT08-032.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 172–183, 2009.
© Springer-Verlag Berlin Heidelberg 2009

is a very popular format to store and integrate a rapidly increasing amount of semi-structured data on the Web while the Semantic Web builds on data represented in RDF and OWL, which is optimized for data interlinking and merging. There exists a wide gap between these data structures, since RDF data (with or without the use of OWL) has domain structure (the concepts and the relationships between concepts) while XML data has document structure (the hierarchy of elements). Also the query languages for these data formats are different. For XML data, XQuery [4] has become the query language of choice, while SPARQL [5] is usually used to query RDF/OWL data.

It would clearly be useful to enable the reuse of RDF data in an XML world and vice versa. Many Web applications have to find a way of querying and processing data represented in XML and RDF/OWL simultaneously. There are several approaches for dealing with *heterogeneous* data consisting of XML, RDF and OWL: The most common approach is to transform all data sources into a single format [6,7] and apply a single query language to this data. Another approach to deal with heterogeneity is query re-writing which poses queries of different query languages to the data which is left in the original format, thus avoiding transformation of the whole data sources [8]. A major drawback of the transformation-based approaches is that the transformation of data from one language into the other is a tedious and error prone process. Indeed, an RDF graph can be represented by more than one XML tree structure, so it is not clear how to formulate XQuery queries against it. On the other hand, XML lacks semantic information; so converting XML to RDF results in incomplete information with a number of blank nodes in the RDF graph. Moreover, many native XML and RDF data storage systems are now available to tackle rapidly increasing data sizes. We expect in the near future that many online RDF/XML sources will not be accessible as RDF/XML files, but rather via data stores that provide a standard querying interface, while the approach of query re-writing limits language functionalities because it is not possible to compile all SPARQL queries entirely into XQuery. In [8], a new query language is designed which allows the formulation of queries on data in different formats. The system automatically generates subqueries in SPARQL and XQuery which are posed to the corresponding data sources in their native format – without the need of data transformation. A major drawback of this approach is that the user has to learn a new query language even though powerful, standardized languages like XQuery and SPARQL exist. Moreover, this approach is not easily extended if data in further formats (like relational data) has to be accessed.

For dealing with *distributed Web data sources*, two major approaches for query processing exist: centralized query processing transfers the distributed data to the central location and processes the query there, while decentralized query processing executes the queries at remote sites whenever this is possible. With the former approach, the data transfer easily becomes the bottleneck of the query execution. Keeping replica on the central location is usually not feasible either, since we are dealing with autonomous and continually updating data sources. Hence, in general, decentralized query processing is clearly superior. Recently DXQ [9] and XRPC [10] have been proposed for decentralized execution of XQuery and, likewise, DARQ [11] for SPARQL. However, to the best of our knowledge, a framework for decentralized query execution to facilitate data integration of heterogeneous Web data sources is still missing.

In this paper we present DeXIN (**D**istributed **e**xtended **X**Query for heterogeneous Data **IN**tegration) – an extensible framework for distributed query processing over heterogeneous, distributed and autonomous data sources. DeXIN considers one data format as the basis (the so-called "aggregation model") and extends the corresponding query language to executing queries over heterogeneous data sources in their respective query languages. Currently, we have only implemented XML as aggregation model and XQuery as the corresponding language, into which the full SPARQL language is integrated. However, our framework is very flexible and could be easily extended to further data formats (e.g., relational data to be queried with SQL) or changed to another aggregation model (e.g., RDF/OWL rather than XML). DeXIN decomposes a user query into subqueries (in our case, XQuery or SPARQL) which are shipped to their respective data sources. These queries are executed at remote locations. The query results are then transformed back into the aggregation model format (for converting the results of a SPARQL query to XML, we adhere to the W3C Proposed Recommendation [12]) and combined to the overall result of the user query. It is important to note that – in contrast to the transformation-based approaches mentioned above [6,7] *only the results are transformed* to a common format. The main contributions of this paper are as follows.

- We present DeXIN – an extensible framework for parallel query execution over distributed, heterogeneous and autonomous large data sources.
- We come up with an extension of XQuery which covers the full SPARQL language and supports the decentralized execution of both XQuery and SPARQL in a single query.
- Our approach supports the data integration of XML, RDF and OWL data without the need of transforming large data sources into a common format.
- We have implemented DeXIN and carried out experiments, which document the good performance and reduced network traffic achieved with our approach.

2 Application Scenario

DeXIN can be profitably applied in any Web environment where large amounts of heterogeneous, distributed data have to be queried and processed. A typical scenario can be the area of Web service management. The number of Web services available for different applications is increasing day by day. In order to assist the service consumer in finding the desired service with the desired properties, several Web service management systems have been developed. The Service Evolution Management Framework (SEMF) [1] [13] is one of these efforts to manage Web services and their related data sources. SEMF describes an information model for integrating the available information for a Web service, keeping track of evolutionary changes of Web services and providing means of complex analysis of Web services. SEMF facilitates the selection of the best Web service from a pool of available Web services for a given task. Each Web service is associated with different attributes which effect the quality of service.

[1] We acknowledge the assistance of Martin Treiber (Distributed Systems Group ,Vienna University of Technology) for providing access to SEMF data.

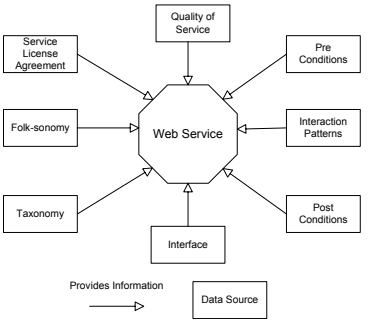


Fig. 1. Data Sources of a Web Service [13]

Figure 1 gives an impression of the diversity of data related to a Web service. This data is normally scattered over various data sources using various languages such XML, RDF, and OWL. However, currently available systems do not treat these heterogeneous, distributed data sources in a satisfactory manner. What is urgently needed is a system which supports different query languages for different data formats, which operates on the data sources as they are without any transformations, and which uses decentralized query processing whenever this is possible. Moreover, this system should be flexible and allow an easy extension to further data formats. In fact, this is precisely the functionality provided by DeXIN.

3 Related Work

Several works are concerned with the transformation of data sources from one language into the other. The W3C GRDDL [6] working group addresses the issue of extracting RDF data from XML files. In [7], SPARQL queries are embedded into XQuery/XSLT and automatically transformed into pure XQuery/XSLT queries to be posed against pure XML data. In great contrast to these two approaches, DeXIN does not apply any transformation to the data sources. Instead, subqueries in SPARQL (or any other language, to which DeXIN is extended in the future) are executed directly on the data sources as they are and only the result is converted. Moreover, in [7], only a subset of SPARQL is supported, while DeXIN allows full SPARQL inside XQuery.

In [8], a new query language XSPARQL was introduced (by merging XQuery and SPARQL) to query both XML and RDF/OWL data. In contrast to [8], our approach is based on standardized query languages (currently XQuery and SPARQL) rather than a newly invented language. Moreover, the aspect of data distribution is not treated in [8].

DXQ[9], XRPC[10] and DARQ[11] are some efforts to execute distributed XQuery and distributed SPARQL separately on XML and RDF data. However, the integration of heterogeneous data sources and the formulation of queries with subqueries from different query languages (like SPARQL inside XQuery) are not addressed in those works.

4 DeXIN

4.1 Architectural Overview

An architectural overview of DeXIN is depicted in Figure 2. The main task of DeXIN is to provide an integrated access to different distributed, heterogeneous, autonomous data sources.

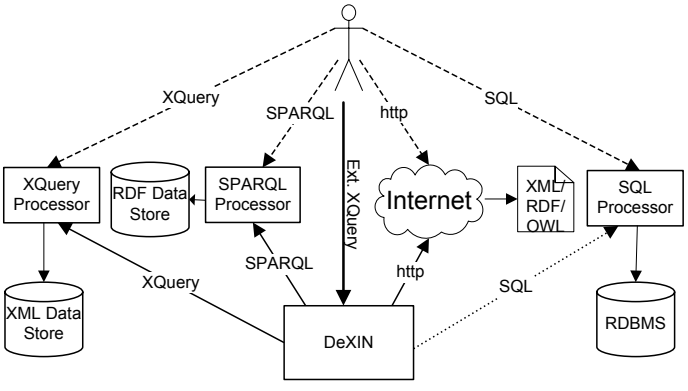


Fig. 2. Architectural overview of DeXIN framework

Normally, the user would have to query each of these data sources separately. With the support of DeXIN, he/she has a single entry point to access all these data sources. By using our extension of XQuery, the user may still formulate subqueries to the various data sources in the appropriate query language. Currently, DeXIN supports XQuery to query XML data and SPARQL to query RDF/OWL data. However, the DeXIN framework is very flexible and we are planning to further extend this approach so as to cover also SQL queries on relational data. Note that not all data sources on the Web provide an XQuery or SPARQL endpoint. Often, the user knows the URI of some (XML or RDF/OWL) data. In this case, DeXIN retrieves the requested document via this URI and executes the desired (XQuery or SPARQL) subquery locally on the site where DeXIN resides. DeXIN decomposes the user query, makes connections to data sources and sends subqueries to the specified data sources. If the execution fails, the user gets a meaningful error message. Otherwise, after successful execution of all subqueries, DeXIN transforms and integrates all intermediate results into a common data format (in our case, XML) and returns the overall result to the user. In total, the user thus issues *a single query* (in our extended XQuery language) and receives *a single result*. All the tedious work of decomposition, connection establishment, document retrieval, query execution, etc. is done behind the scene by DeXIN.

4.2 Query Evaluation Process

The query evaluation process in DeXIN is shown in Figure 3. The main components of the framework are briefly discussed below.

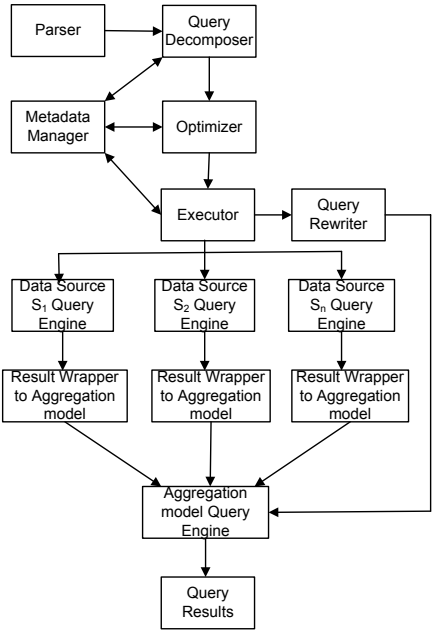


Fig. 3. Query Evaluation Process

Parser. The Parser checks the syntax of the user query. If the user query is syntactically correct, the parser will generate the query tree and pass it on to the query decomposer. Otherwise it will return an error to the user.

Query Decomposer. The Query Decomposer decomposes the user query into atomic subqueries, which apply to a single data source each. The concrete data source is identified by means of the information available in the Metadata Manager (see below). Each of these atomic subqueries can then be executed on its respective data source by the Executor (see below).

Metadata Manager. All data sources supported by the system are registered by the Metadata Manager. For each data source, the Metadata Manager contains all the relevant information required by the Query Decomposer, the Optimizer or the Executor. Metadata Manager also stores information like updated statistics and availability of data sources to support the Optimizer.

Optimizer. Optimizer searches for the best query execution plan based on static information available at the Metadata Manager. It also performs some dynamic optimization to find variable dependencies in the dependant or bind joins. Dependant or bind joins are basically nested loop joins where intermediate results from the outer relation are passed as filter to the inner loop. Thus, for each value of a variable in the outer loop, a new subquery is generated for execution at the remote site. In such scenarios, the optimizer will first look for all possible values of the variables in the outer loop and ground the variables in the subquery with all possible values, thus formulating a bundled query to ship at once to the remote site.

Executor. The Executor schedules the execution sequence of all the queries (in parallel or sequential). In particular, the Executor has to take care of any dependencies between subqueries. If a registered data source provides an XQuery or SPARQL endpoint, then the Executor establishes the connection with this data source, issues the desired subqueries and receives the result. If a registered data source only allows the retrieval of XML or RDF/OWL documents via the URI, then the Executor retrieves the desired documents and executes the subqueries locally on its own site. Of course, the execution of a subqueries may fail, e.g., with source unreachable, access denied, syntax error, query timeout, etc. It is the responsibility of the Executor to handle all these exceptions. In particular, the Executor has to decide if a continuation makes sense or the execution is aborted with an error message to the user.

Result Reconstruction. All the results received from distributed, heterogeneous and federated data sources are wrapped to the format of the aggregation model (in our case, XML). After wrapping the results, this component integrates the results and stores them in temporary files for further querying by the aggregation model query processor (in our case, an XQuery engine).

Query Rewriter. The Query Rewriter rewrites the user query in the extended query language (in our case, extended XQuery) into a single query on the aggregation model (in our case, this is a proper XQuery query which is executed over XML sources only). For this purpose, all subqueries referring to different data sources are replaced by a reference to the locally stored result of these subqueries. The overall result of the user query is then simply obtained by locally executing this rewritten query.

5 XQuery Extension to SPARQL

DeXIN is an extensible framework based on a multi-lingual and multi-database architecture to deal with various data formats and various query languages. It uses a distinguished data format as "aggregation model" together with an appropriate query language for data in this format. So far, we are using XML as aggregation model and XQuery as the corresponding query language. This aggregation model can then be extended to other data formats (like RDF/OWL) with other query languages (like SPARQL). In order to execute SPARQL queries inside XQuery, it suffices to introduce a new function called SPARQLQuery(). This function can be used anywhere in XQuery where a reference to an XML document may occur. This approach is very similar to the extension of SQL via the XMLQuery function in order to execute XQuery inside SQL (see [14]). The new function SPARQLQuery() is defined as follows:

```
XMLDOC SPARQLQuery(String sparqlQuery,URI sourceURI)
```

The value returned by a call to this function is of type *XMLDOC*. The function SPARQLQuery() has two parameters: The first parameter is of type *String* and contains the SPARQL query that has to be executed. The second parameter is of type *URI* and either contains the URI or just the name of the data source on which the SPARQL query has to be executed. The name of the data source refers to an entry in the database of known

data sources maintained by the Metadata Manager. If the indicated data source is reachable and the SPARQL query is successfully executed, then the result is wrapped into XML according to the W3C Proposed Recommendation [12].

To illustrate this concept, we revisit the motivating example of SEMF[13] discussed in Section 3. Suppose that a user wants to get information about available Web services which have a license fee of less than one Euro per usage. Moreover, suppose that the user also needs information on the service license agreement and the quality of service before using this service in his/her application. Even this simple example may encounter the problem of heterogeneous data sources if, for example, the service license agreement information is available in XML format while the information about the quality of service is available in RDF format. A query in extended XQuery for retrieving the desired information is shown in Figure 4.

```
for
  $a in doc("http://SEMF/License.xml")/agreement,
  $b in SPARQLQuery(" SELECT ?title ?ExecutionTime
   WHERE {
?x <http://www.w3.org/2001/sub#title> ?title .
?x  <http://www.w3.org/2001/sub#QoS>   ?ExecutionTime "
          },http://SEMF/QoS.rdf)/result
WHERE
$a/servicetitle = $b/title
AND $a/peruse/amount <= 1
RETURN
<Results>
 <Service>
  <ServiceTitle>{$a/title}</ServiceTitle>
  <Requirement>{$a/requirement}</Requirement>
  <ExecutionTime>{$b/ExecutionTime}</ExecutionTime>
 </Service>
</Results>
```

Fig. 4. An example extended XQuery for DeXIN

We conclude this section by having a closer look at the central steps for executing an extended XQuery query, namely the query decomposition and query execution.

The query tree returned by the Parser has to be traversed in order to search for all calls of the SPARQLQuery() function. Suppose that we have n such calls. For each call of this function, the Query Decomposer retrieves the SPARQL query q_i and the data source d_i on which the query q_i shall be executed. The result of this process is a list $\{(q_1, d_1), \ldots, (q_n, d_n)\}$ of pairs consisting of a query and a source. The Executor then poses each query q_i against the data source d_i. The order of the execution of these queries and possible parallelization have to take the dependencies between these queries into account. If the execution of each query q_i was successful, its result is transferred to the site where DeXIN is located and converted into XML-format. The resulting XML-document r_i is then stored temporarily. Moreover, in the query tree received from the Parser, the call of the SPARQLQuery() function with query q_i and data source d_i is replaced by a reference to the XML-document r_i. The resulting query tree is a query tree of pure XQuery without any extensions. It can thus be executed locally by the XQuery engine used by DeXIN.

6 Implementation and Experiments

DeXIN supports queries over distributed, heterogeneous and autonomous data sources. It can be easily plugged into applications which require such a facility. As a case study, we take the example of service management systems and show how DeXIN enhances service management software by providing this query facility over heterogeneous and distributed data sources. We set up a testbed which includes 3 computers (Intel(R) Core(TM)2 CPU, 1.86GHz, 2GB RAM) running SUSE Linux with kernel version 2.6. The machines are connected over a standard 100Mbit/S network connection. An open source native XML database eXist (release 1.2.4) is installed on each system to store XML data. Our prototype is implemented in Java. We utilize the eXist [15] XQuery processor to execute XQuery queries. The Jena Framework [16] (release 2.5.6) is used for storing the RDF data, and the ARQ query engine packaged within Jena is used to execute SPARQL queries.

6.1 Experimental Application: Web Service Management

One of the main motivations for developing this framework is to utilize it for service management systems like SEMF [13]. Being able to query distributed and heterogeneous data sources associated to Web services is a major issue in these systems. SEMF stores and manages updated information about all the services listed in this framework. Recall the example use case given in Section 5: We consider a user who requests information about available Web services which have a license fee of less than one Euro per usage. Moreover, the user needs information on the service license agreement and the quality of service. We assume that the service license agreement information is available in XML format while the information about the quality of service is available in RDF format. As we have seen in Section 5, our framework provides the user a convenient way of querying these distributed, heterogeneous data sources at the SEMF platform without worrying about the transformation, distribution and heterogeneity of the data sources involved by issuing the extended XQuery query of Figure 4 to SEMF. The result returned to the user is in XML format and may look like the XML file in Figure 5.

6.2 Performance Analysis

In order to analyze the performance of DeXIN, we have conducted tests with realistically large data. Since SEMF is only available as a prototype, the test data available in this context is too small for meaningful performance tests. We therefore chose to use DBPedia (see http://dbpedia.org/) and DBLP (see http://dblp.uni-trier.de/xml/), which are commonly used for benchmarking.

Data Distribution over the Testbed. For the SPARQL query execution over RDF data, we use a subset of DBPedia, which contains RDF information extracted from Wikipedia. This data consists of about 31.5 million triples and is divided into three parts (Articles, Categories, Persons). The size of these parts is displayed in Table 1. The data is distributed over the testbed in such a way that the Articles, Categories, and Persons are stored on different machines. Moreover, we have further split these data sets into 10 data sources of varying size in order to formulate queries with subqueries

```
<Results>
 <Service>
  <ServiceTitle>WISIRISFuzzySearch</ServiceTitle>
  <Requirement>
   <peruse>
    <payment>
     <amount currency='EUR'> 0.90 </amount>
     <taxpercent code='VAT'>20</taxpercent>
    </payment>
   </peruse>
  </Requirement>
  <ExecutionTime Unit='sec'>17</ExecutionTime>
 </Service>
 <Service>
    .........
 </Service>
  ......
</Result>
```

Fig. 5. Result after Executing the Query shown in Figure 4

for a bigger number of data sources. For the XQuery execution over XML data we used DBLP. DBLP is an online bibliography available in XML format, which lists more than 1 million articles. It contains more than 10 million elements and 2 million attributes. The average depth of the elements is 2.5. The XML data is also divided into three parts (Articles, Proceedings, Books), whose. size is shown in Table 2. We distributed the XML data over the testbed such that the Articles, Proceedings, and Books are stored on different machines. As with the RDF data, we also subdivided each of the three parts of the XML data into several data sources of varying size.

Table 1. RDF Data Sources

Name	Description	# Tuples
RS1	Articles	7.6Million
RS2	Categories	6.4Million
RS3	Persons	0.6Million

Table 2. XML Data Sources

Name	Description	Size
XS1	Articles	250MB
XS2	Proceedings	200MB
XS3	Books	50MB

Experiments. In the first scenario we consider a set of queries of different complexity varying from simple select-project queries to complex join queries. The queries use a different number of distributed sources and have different result sizes. The results shown are the average values over ten runs. The query execution time is subdivided as

Total Time = Connection Time + Execution Time + Transfer Time

Figure 6 presents the query execution time for a naive centralized approach compared with DeXIN. It turns out that the data transfer time is the main contributor to the query execution time in the distributed environment – which is not surprising according to the theory on distributed databases [17]. DeXIN reduces the amount of data transferred over the network by pushing the query execution to the local site, thus transferring only the query results. We observe that with increasing size of data sets, the gap in the query execution time between DeXIN and the naive centralized approach is widened.

In the second scenario we fix the size of data sources and execute queries with varying selectivity factor (i.e., the ratio of result size to data size) and compare the query

execution time of DeXIN with the naive centralized approach. As was already observed in the previous scenario, the execution time is largely determined by the network transfer. Figure 7 further strengthens this conclusion and, moreover, shows that DeXIN gives a better execution time for queries with high selectivity. The results displayed in Figure 7 indicate that DeXIN is much stronger affected by varying the selectivity of queries than the centralized approach. DeXIN is superior to the centralized approach as long as the selectivity factor is less than 90% . Above, the two approaches are roughly equal.

In the third scenario, we observe the effect of the number of data sources on the query execution time. We executed several queries with varying number of sources used in each query. Figure 8 again compares the execution time of DeXIN with the execution time of a naive centralized approach. It turns out that as soon as the number of sources exceeds 2, DeXIN is clearly superior.

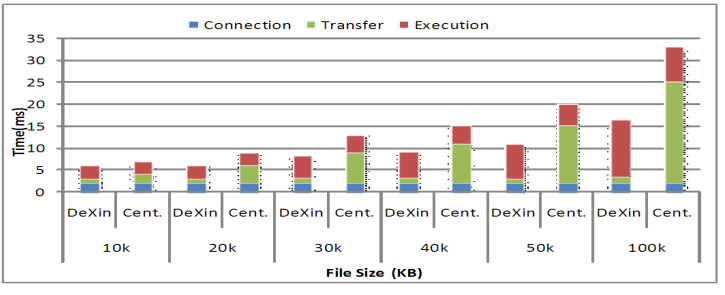


Fig. 6. Execution Time Comparison

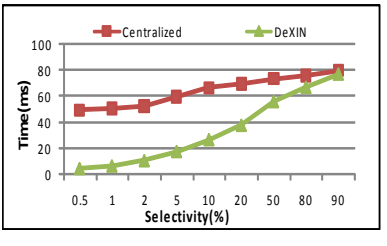


Fig. 7. Varying Selectivity Factor

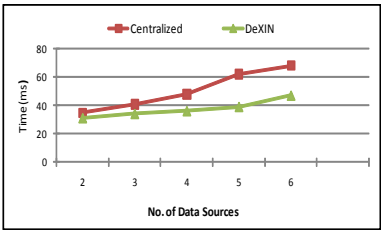


Fig. 8. Varying level of Distribution

7 Conclusions and Future Work

In this paper, we have presented DeXIN – a novel framework for an integrated access to heterogeneous, distributed data sources. So far, our approach supports the data integration of XML and RDF/OWL data without the need of transforming large data sources into a common format. We have defined and implemented an extension of XQuery to provide full SPARQL support for subqueries. It is worth mentioning that the XQuery extension not only enhances XQuery capabilities to execute SPARQL queries, but SPARQL is also enhanced with XQuery capabilities e.g. result formatting in the return clause of XQuery etc.

DeXIN can be easily integrated in distributed web applications which require querying facility in distributed or peer to peer networks. It can become a powerful tool for knowledgeable users or web applications to facilitate querying over XML data and reasoning over Semantic Web data simultaneously.

An important feature of our framework is its flexibility and extensibility. A major goal for future work on DeXIN is to extend the data integration to further data formats (in particular, relational data) and further query languages (in particular, SQL). Moreover, we are planning to incorporate query optimization techniques (like semi-joins – a standard technique in distributed database systems [17]) into DeXIN. We also want to extend the tests with DeXIN. So far, we have tested DeXIN with large data sets but on a small number of servers. In the future, when the Web service management system SEMF [13] is eventually applied to realistically big scenarios, DeXIN will naturally be tested in an environment with a large-scale network.

References

1. Bray, T., Paoli, J., Sperberg-McQueen, C.M., Maler, E., Yergeau, F.: Extensible Markup Language (XML) 1.0, 4th edn., W3C Proposed Recommendation (September 2006)
2. Beckett, D., McBride, B.: RDF/XML Syntax Specification (Revised), W3C Proposed Recommendation (February 2004)
3. McGuinness, D.L., van Harmelen, J.: OWL Web Ontology Language. W3C Proposed Recommendation (February 2004)
4. Boag, S., Chamberlin, D., Fernández, M.F., Florescu, D., Robie, J., Siméon, J.: XQuery 1.0: An XML Query Language, W3C Proposed Recommendation (January 2007)
5. Prud'hommeaux, E., Seaborne, A.: SPARQL Query Language for RDF, W3C Proposed Recommendation (January 2008)
6. Gandon, F.: GRDDL Use Cases: Scenarios of extracting RDF data from XML documents. W3C Proposed Recommendation (April 2007)
7. Groppe, S., Groppe, J., Linnemann, V., Kukulenz, D., Hoeller, N., Reinke, C.: Embedding sparql into xquery/xslt. In: Proc. SAC 2008, pp. 2271–2278 (2008)
8. Akhtar, W., Kopecký, J., Krennwallner, T., Polleres, A.: Xsparql: Traveling between the xml and rdf worlds - and avoiding the xslt pilgrimage. In: Bechhofer, S., Hauswirth, M., Hoffmann, J., Koubarakis, M. (eds.) ESWC 2008. LNCS, vol. 5021, pp. 432–447. Springer, Heidelberg (2008)
9. Fernández, M.F., Jim, T., Morton, K., Onose, N., Siméon, J.: Highly distributed xquery with dxq. In: SIGMOD Conference, pp. 1159–1161 (2007)
10. Zhang, Y., Boncz, P.A.: Xrpc: Interoperable and efficient distributed xquery. In: VLDB, pp. 99–110 (2007)
11. Quilitz, B., Leser, U.: Querying distributed rdf data sources with sparql. In: Bechhofer, S., Hauswirth, M., Hoffmann, J., Koubarakis, M. (eds.) ESWC 2008. LNCS, vol. 5021, pp. 524–538. Springer, Heidelberg (2008)
12. Beckett, D., Broekstra, J.: SPARQL Query Results XML Format, W3C Proposed Recommendation (January 2008)
13. Treiber, M., Truong, H.L., Dustdar, S.: Semf - service evolution management framework. In: Proc. EUROMICRO 2008, pp. 329–336 (2008)
14. Melton, J.: SQL, XQuery, and SPARQL: What's Wrong With This Picture? In: Proc. XTech (2006)
15. Meier, W.M.: eXist: Open Source Native XML Database (June 2008)
16. Jena: A Semantic Web Framework for Java (June 2008)
17. Özsu, M.T., Valduriez, P.: Principles of Distributed Database Systems. Prentice-Hall, Englewood Cliffs (1999)

Dimensional Templates in Data Warehouses: Automating the Multidimensional Design of Data Warehouse Prototypes

Rui Oliveira[1], Fátima Rodrigues[2], Paulo Martins[3], and João Paulo Moura[3]

[1] Dep. Engenharia Informática, Escola Superior de Tecnologia e Gestão
Instituto Politécnico de Leiria, 2411-901 Leiria, Portugal
rui.oliveira@estg.ipleiria.pt
[2] GECAD – Grupo de Investigação em Engenharia do Conhecimento e Apoio à Decisão
Dep. Engenharia Informática, Instituto Superior de Engenharia do Porto, Porto, Portugal
mfc@isep.ipp.pt
[3] GECAD – Grupo de Investigação em Engenharia do Conhecimento e Apoio à Decisão
Universidade de Trás-os-Montes e Alto Douro, Vila Real, Portugal
{pmartins,jpmoura}@utad.pt

Abstract. Prototypes are valuable tools in Data Warehouse (DW) projects. DW prototypes can help end-users to get an accurate preview of a future DW system, along with its advantages and constraints. However, DW prototypes have considerably smaller development time windows when compared to complete DW projects. This puts additional pressure on the achievement of the expected prototypes' high quality standards, especially at the highly time consuming multidimensional design: in it, a thin margin for harmful unreflected decisions exists. Some devised methods for automating DW multidimensional design can be used to accelerate this stage, yet they are more suitable to DW full projects rather than to prototypes, due to the effort, cost and expertise they require. This paper proposes the semi-automation of DW multidimensional designs using templates. We believe this approach better fits the development speed and cost constraints of DW prototyping since templates are pre-built highly adaptable and highly reusable solutions.

Keywords: Data Warehouse, Automated Multidimensional Design, Dimensional Templates, Prototype Development.

1 Introduction

Prototypes are valuable tools in DW projects and much as been written on the subject, not only by field practitioners [1,2] but also by the scientific community [3], to mention only a few. A first benefit of DW prototypes is that they act as a preview of the satisfiable end-users' requirements considering the available data sources. This avoids later costly disappointments about the DW outcome. Secondly, DW prototypes allow predicting with a high degree of confidence the restraining factors on the future

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 184–195, 2009.
© Springer-Verlag Berlin Heidelberg 2009

DW project, such as cost, size or deadlines. Finally, DW prototypes are ideal to materialize the benefits of future DW projects, thus helping to justify the overall investment.

DW prototypes have shorter development time windows and considerably more restrained budgets when compared to full DW projects. However, the urge to develop a DW prototype cannot endorse a low quality product. In fact, a DW prototype must be more than just a DW sketch in which design and implementation errors will vanish once the prototype is thrown away. Quite the opposite, a DW prototype must be built as a launching platform to a full-scale DW project once it has proven its point [2,3]. Therefore, if it is reasonable to expect that a DW prototype can be incomplete due to time and cost restrictions, it is less so in what concerns accuracy.

Accuracy and fast development are not easy to balance goals in DWs. This happens due to the many sensible phases requiring well-reflected decisions, such as the multidimensional design stage. Accurate multidimensional design consumes a considerable amount of time and human expert resources [4,5]: business requirements must be gathered, data sources must be deeply analysed, DW expertise must be acquired, and performance plus storage sustainability must be assured. Such tasks do not fully comply with the pressure of time, especially in highly time-restrained environments such as DW prototypes.

Aiming to accelerate the multidimensional design stage of DWs, some semi-automated methods have been devised, such as [6,7,8], among others. Although effective and justifiable in the context of a complete DW system development, these methods' requirements can be inadequate for most DW prototypes. This happens because such methods need a deep understanding of data sources by DW designers, solid multidimensional design expertise or even specific data source documentation. Such demands consume time and budget resources not likely attainable in many DW prototype projects.

It is our belief that DW prototyping provides the perfect conditions for introducing the use of generic multidimensional solutions rather than personalized ones. In that sense, the current paper proposes the innovative use of templates as a way of performing multidimensional design in DW prototypes. This approach aims to solve some key problems that effectively delay this development stage. In fact, templates are a widely accepted mechanism in many informatics areas with the purpose of accelerating software development speed. Even though the resulting product turns out to be neither personalized nor optimised, its structure has a quality guaranty and allows additional refinements. Also, the use of templates avoids the need to gather expert knowledge in order to perform common tasks. Finally, templates are generic and therefore adaptable to the needs of a wide number of particular scenarios. From our perspective, such positive qualities can be successfully adapted to the DW prototyping domain, as addressed in this paper.

The paper is structured as follows. Section 2 briefly presents the state of the art on multidimensional design semi-automation methods, template usage and DW prototyping tools. Section 3 describes the overall concept of the proposed approach and the construction of dimensional templates. Section 4 presents the algorithm for generating multidimensional structures from dimensional templates. Finally, section 5 concludes the paper.

2 Related Work

In the past, valuable methods have been devised to semi-automate the multidimensional design of DWs, with focus on diminishing the time consumed while maintaining high quality standards. Some of these methods provide semi-automated multidimensional design based on specific formats such as E/R schemes [9], XML schemes [6], object oriented [10,11] or even oriented to ontologies [12]. At some extent, such methods help reducing DW projects' development time and DW experts' involvement. The main drawback in using these powerful techniques is that they require source data to be well documented using a specific format, like E/R schemes, UML diagrams or even an ontology language. Even though these are scientifically accepted specifications, many documentation problems may easily rise concerning source data's documentation [13]: poor maintenance, physical implementations differing from logical designs, the use of non-standard formats and, the worst case of all, no documentation at all.

Other multidimensional semi-automated methods avoid the need of specific documentation formats to describe data relationships and processes at source systems. For instance, [8] proposes a method for multidimensional design automation by applying a generation algorithm to data sources previously tagged with multidimensional markers. To be effective, the method simultaneously requires a thorough inspection of data sources and the existence of dimensional expertise from the ones in charge of that task: otherwise, the resulting multidimensional design can be far from correctness. Also, the approach assumes that a third normal form database is used, not accounting for cases in which a high degree of denormalization is found. Again, it is a valuable method assuming that the available time window for multidimensional design is comfortable and that the cost of getting DW experts' support is acceptable at the early prototyping phase.

Another known multidimensional semi-automated method based on the analysis of data sources without the need of specific documentation is [14]. In it, reverse engineering is used to obtain relational metadata. We consider the method not the best choice for DW prototyping, since it is not suitable for large and complex systems. Also, the method becomes hard to apply since it uses non-standard modelling techniques.

Also worth mentioning is the method proposed in [7], which emphasises the use of end-user requirement driven multidimensional design. Although powerful to validate end-user requirements against organizational data in an automated way, it requires a strong interaction between DW design experts and end-users. Also, it requires that organizations' processes be specified using the method's particular format.

As concerns the use of templates, they are a widely accepted approach in the informatics area for automating operations. The range of applications goes from the most simple office software to highly demanding industrial tools. That we know of, there are no proposals of template usage to automate the construction of multidimensional designs. Authors in the literature of multidimensional design, such as [15,16] among others, present techniques, guidelines and standard models for building multidimensional designs from scratch. However, their proposals cannot be seen as templates but rather as theoretical models requiring significant DW expertise to be understood and adapted.

As concerns software tools dedicated to DW prototypes' management, some choices are available at the time of this writing, like [3, 17]. However, today's

Extract- Transform-Load (ETL) tools can be effectively used to prototype DW systems, ranging from open source to commercial tools. [18] gives an extensive list of ETL tools, even though others exist. Common to the generality of DW prototyping tools (dedicated or standard ETL) is the lack of support given to the automation of multidimensional designs, which is the context of the proposed approach. Although such tools can support the crucial implementation and maintenance phases of DW prototypes, a global assumption is that a previously devised multidimensional design already exists.

3 Dimensional Templates

In this paper we propose the use of templates to automate the multidimensional design phase of DW prototypes. To distinguish the here-proposed templates from other areas' templates, ours are named *dimensional templates*.

Along this paper, the case study of a retail sales company is used to illustrate the proposed concepts, particularly its business process *retail sales* [15].

3.1 The Overall Concept

Fig. 1 depicts the three stages through which a dimensional template can go through, described as follows:

Construction. End-user requirements (EURs) concerning generic distinct business processes (like *retail sales* or *inventory levels*) are represented using logical models named *rationale diagrams*. The set of rationale diagrams concerning a specific business process constitutes a *dimensional template*. This stage, to be conducted by DW design experts, is analysed in section 3.

Acquisition. This stage is conducted from inside the organization requiring the multidimensional design for a DW prototype. After gathering the EURs for the DW prototype, the necessary dimensional templates are acquired. This stage requires no DW knowledge.

Configuration. Dimensional templates are suitable to a non-limited number of real scenarios. In order to generate a multidimensional design for an organization's particular scenario, the dimensional templates gathered at the *acquisition stage* need to be configured and latter processed by a generation algorithm. This stage, which requires no DW knowledge, is further analysed in section 4.

3.2 Building Dimensional Templates

As mentioned, dimensional templates are composed of rationale diagrams representing generic EURs for a particular DW business process. Concerning the formal representation of EURs, much has been written. Existing methods, like [7,19], extend the original *i** framework specifically for DW development. The work of [7] was found to be the most useful for our approach, due to its simple notation. From it, some basic elements were imported. These are as follows:

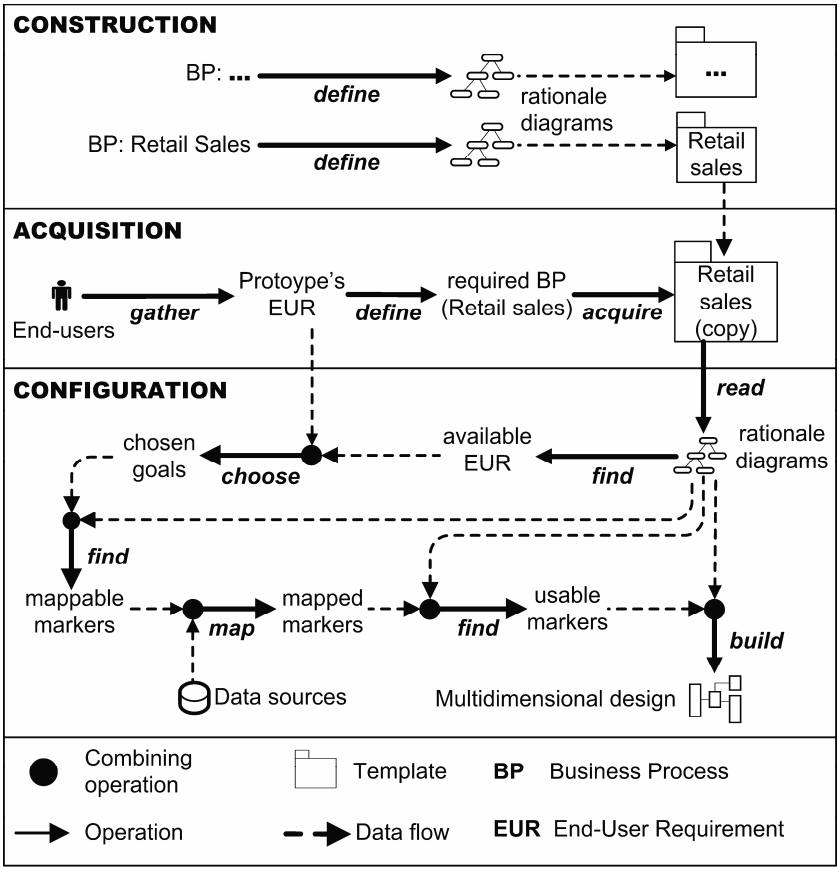


Fig. 1. Overall view of the approach using the case study of a retail-sales company requiring a DW prototype

Goal. Represents an EUR. A goal can be decomposed into more specific child goals representing more detailed versions of their parent goal.

Decomposition. Represents the division of a parent goal into several child goals. A decomposition can be an AND-decomposition (every child goal must be satisfied so that its parent goal is satisfied) or an OR-decomposition (at least one child goal must be satisfied so that its parent goal is satisfied).

Rationale Diagram. Logical representation of an EUR with all its decompositions. In its original form [7], rationale diagrams are used to relate EURs with *actors* and *facts*, but these two concepts were not imported.

In the context of our approach we have extended the concepts of *goal*, *decomposition* and *rationale diagram* with new key elements, as described next.

Table 2 depicts the graphical notation of our rationale diagrams' elements.

Grain-goal. Representation of a goal in the context of a specific grain (the data's granularity). The grains associated to *grain-goals* are the ones considered as being *reasonable* for the specific business process. Table 1 shows some examples of grains described as reasonable. This means that other grains may exist, yet the amount of data required to satisfy them would become unmanageable (like a *bit* grain for the Network Cable Company's scenario). Since the number of reasonable grains for each business process is small, the amount of possible grain-goals for each child goal does not compromise rationale diagrams' manageability.

Table 1. Notation used in the proposed rationale diagrams

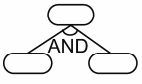
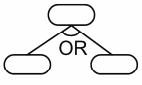
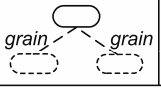
Symbol	Meaning	Symbol	Meaning
	goal		AND-decomposition
	grain-goal		
	dimensional context		OR-decomposition
	marker		grain-decomposition

Table 2. Reasonable grains for two distinct scenarios

Scenario	Business Process	Reasonable Grains
Retail Sales Company	Retail Sales	Sale Line-of-sale
	Inventory Levels	Periodic snapshot Transaction
Network Cable Company	Customer Billing	Bill Customer session Packet
	Network Traffic	Packet

Grain-decomposition. Represents the division of a goal into as many grain-goals as the number of reasonable grains.

Marker. The representation of a type of data required to exist in data source systems so that a specific grain-goal can be satisfied.

Dimensional Context. The information context into which a specific marker fits. Information contexts are detailed further on.

3.3 Rationale Diagrams

Each rationale diagram, as presented in this paper, is a logical representation of the source systems' data required to satisfy a certain EUR, that is, a logical mapping of goals to markers. Fig. 2 depicts a rationale diagram for the goal *Analyse product sales*

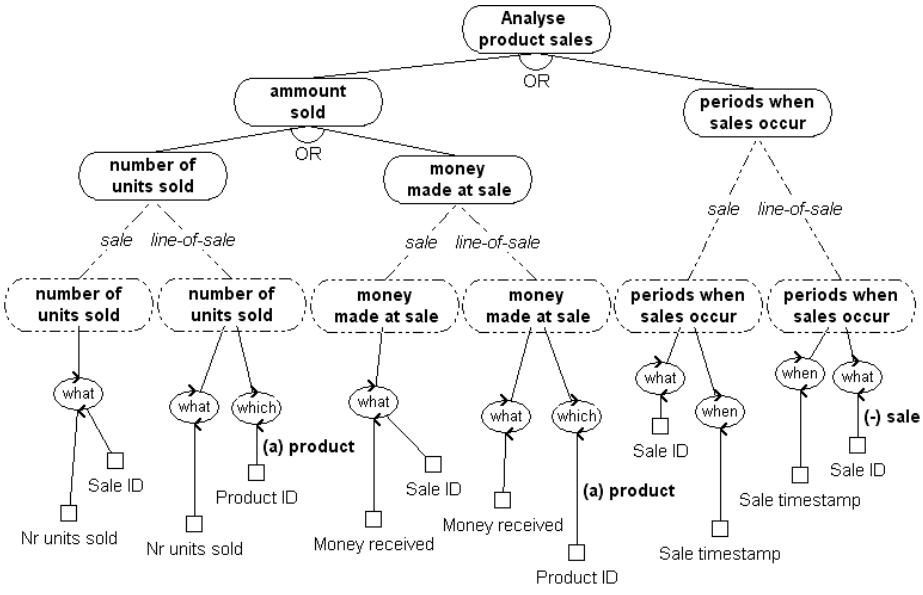


Fig. 2. Simplified rationale diagram for the EUR Analyse product sales in the retail sales company case study

(simplified for clarity sake) concerning the retail sales company case study. As shown, the parent goal is divided into more detailed versions (child goals) using OR-decompositions. At the lowest level of each OR-decomposition, a final division into grain-goals is performed. In rationale diagrams, in general, if no AND/OR decomposition is considered adequate for a goal, a grain-decomposition is applied.

Dimensional Contexts. The primary role of multidimensional structures is to enhance the ability of end-users to answer the question *why did facts occur in the source system?*. The *why* question can be decomposed into one *f*-question and five *d*-questions, each aiming to clarify facts occurrence in distinct information perspectives.

The *f*-question is *what happened in the source systems?* and it is answered by the facts and measures found in fact tables. The *d*-questions can be answered using dimension tables plus their foreign keys' links to fact tables, and are as follows:

- *How* did facts took place (what where the environmental conditions when facts occurred? E.g., promotions, discounts);
- *When* did facts occur (time);
- *Where* did facts took place (e.g., store, web, warehouse);
- *Which* agents passively participated in facts' occurrence (e.g., product, web page);
- *Who* did actively motivated the facts' occurrence (e.g., salesman, customer).

As concerns our approach, we have defined the concept of *dimensional context*, representing the informative context to which any multidimensional structure relates. Therefore, six dimensional contexts can be found in a multidimensional schema: *how*, *what*, *when*, *where*, *which* and *who*.

Markers and Dimensional Contexts. At this point, four assumptions (*A*) can be made and a corollary (*C*) can be derived: (*A1*) EURs are satisfied by multidimensional structures and their data; (*A2*) a multidimensional structure is always related to a dimensional context; (*A3*) the data contained in a multidimensional structure shares the structure's dimensional context; (*A4*) the data contained in a multidimensional structure is transformed/cleansed data originating from source systems; (*C*) every source data element which satisfies an EUR has a dimensional context (and so will the marker representing such data element).

Fig. 2 helps illustrating the previous corollary: the marker *Product ID* linked to the grain-goal *number of units sold* clearly belongs to *which* context, since it refers to something that passively participates in facts' occurrence (e.g., products). Analysing the same grain-goal, the marker *nr units sold* answers no *d*-question: then, by default, it answers the *f*-question (thus relating to the *what* context).

Tagging Markers. A marker represents a type of source systems' data required to satisfy grain-goals. Generally, the grain of a marker's data is the same as the grain-goal's grain it relates to. For instance, Fig. 2 shows that to satisfy the grain-goal *number of units sold* at the grain level *line-of-sale*, the number of units sold for each line-of-sale is required (marker *nr units sold*).

However, grain-goals may eventually require markers with a lower grain level than its own (*sale* grain is considered lower than *line-of-sale* grain because it supports less detailed data). Analysing Fig. 2, the grain-goal *periods when sales occur* at the line-of-sale grain level is satisfiable with the *Sale ID* marker, which relates to sales, while the grain-goal refers to lines-of-sale (different grain levels). This exceptions may occur with markers related to *when* and *what* dimensional contexts. Once detected, they are dealt by tagging the corresponding marker-dimensional context connection with the *(-)* symbol followed by the name of the grain the marker refers to.

As concerns markers related to *how*, *where*, *which* and *who* dimensional contexts, it is important to mention the business agent involved in the grain-goal's satisfaction. An agent is a source systems' physical actor or event to which a marker refers. For instance, in the addressed case study, common agents for *which*-related markers are *product*, while for *who*-related markers two common agents are *customer* and *employee*. Agents are represented in rationale diagrams by tagging the corresponding marker-dimensional context connection with the *(a)* symbol followed by the agent's name (see Fig. 2 for some examples of tagging with the *product* agent).

4 Using Dimensional Templates

In this section we briefly present the algorithm for generating multidimensional designs from rationale diagrams (Fig. 1, configuration stage). Some screen captures of a template configuration tool prototype developed by the authors are used to illustrate the several steps of the generation algorithm. It is worth mentioning that the theoretical concepts used to build the algorithm are the widely accepted ones of [15].

Consider following Fig. 1 (configuration stage) and Fig. 2 for a better understanding of the algorithm's explanation, since the case study of retail sales will also be used in this section. A series of definitions (*D*) will be used throughout the algorithm's explanation.

At this point, it will be assumed that the dimensional templates required to satisfy the business processes found at the acquisition stage have been gathered. Such templates include rationale diagrams containing DW EURs in the form of goals.

4.1 Step 1: Finding Mappable Markers

The generation algorithm will not use all the goals contained inside dimensional templates, but only those who match the EURs defined at the acquisition stage (*D1: chosen goal*). The algorithm then accesses the template's rationale diagrams to determine which markers must be mapped in order to satisfy each of the chosen goals (*D2: mappable marker*). Fig. 3 depicts the mappable markers for the chosen goal *Money made at sale* (also visible in Fig. 2) for each of its grain-goals.

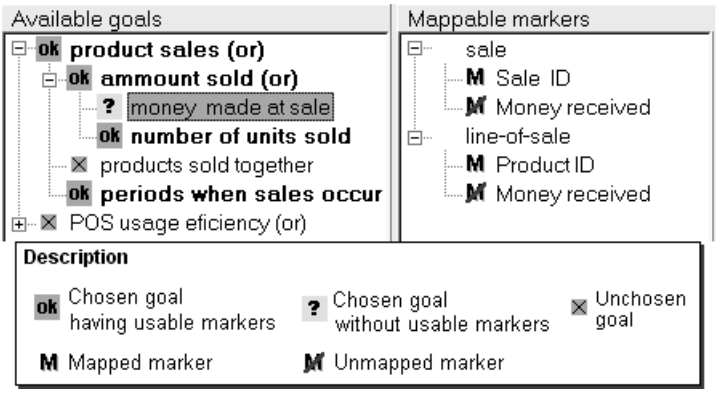


Fig. 3. Partial screen capture of a template configuration tool showing the *chosen goals* and their *mappable markers* at each grain, which can also be seen in the rationale diagram of Fig. 2

4.2 Step 2: Mapping Markers

Mappable markers are useless until they are mapped to real source data. A mapped marker consists of a marker to which the correct physical location of data has been provided (*D3: mapped marker*). This mapping operation is important to define the logical data map [20] after multidimensional structures have been generated.

4.3 Step 3: Determining Usable Markers

If all of a grain-goal's markers are mapped, those markers will be considered *usable* (*D4: usable marker*). The algorithm will only consider for multidimensional generation the usable markers found. In Fig. 3 it is visible that the goal *Money made at sale* has unmapped markers at all grains. This means that none of its markers is *usable* (the *Sale ID* and *Product ID* markers, although mapped, are not usable).

4.4 Step 4: Determining Satisfied Goals

A grain-goal is considered satisfied if it only contains usable markers (*D5: satisfied grain-goal*). A goal having child goals will be considered satisfied if (i) an AND-decomposition

is used and *all* of its child goals are satisfied or if (ii) an OR-decomposition is used and *at least one* of its child goals is satisfied (*D6: satisfied goal*). A goal having only child grain-goals, like *Money made at sale*, will be considered satisfied if it has *at least one* satisfied grain-goal.

4.5 Step 5: Multidimensional Generation

According to [15], different grains need to be addressed by separate fact tables and therefore by distinct multidimensional designs (*star-schemas*). Accordingly, our generation algorithm must be able to generate as many distinct star-schema models as the number of reasonable grains for which satisfied grain-goals exist (*D5*). In order to do so, the algorithm must be run one time for each reasonable grain (an algorithm iteration). Each iteration thus refers to a single grain (*D7: iteration grain*) and will generate its own fact table. For each algorithm's iteration, the steps are as follows:

1. With usable markers linked to *what* or *when* dimensional contexts and linked to grain-goals with the same grain as the iteration's grain, finds all distinct trios <*marker*, *marker's dimensional context*, *marker's grain*>. From Fig. 2 goal *periods when sales occur* at the line-of-sale grain, the retrieved trio for the iteration grain line-of-sale is <*Sale ID*, *what*, *sale*>. For each distinct trio found:
   - 1.1 If the dimensional context is *when*, time related multidimensional elements are required:
     - 1.1.1 If the marker's grain is the same as the iteration's grain, a foreign key is created between the iteration's fact table and the time dimension.
     - 1.1.2 If the marker's grain is lower than the iteration's grain then a measure is created in the fact table, using the marker's name.
   - 1.2 If the dimensional context is *what*, a fact table related element is necessary:
     - 1.2.1 If the marker's grain is the same as the iteration's grain then a measure is created in the fact table, using the marker's name.
     - 1.2.2 If the marker's grain is lower than the iteration's grain, a degenerated dimension is created in the fact table, using the marker's name.
2. With usable markers linked to *how*, *where*, *which* or *who* dimensional contexts and linked to grain-goals with the same grain as the iteration's grain, finds all distinct trios <*marker*, *marker's dimensional context*, *marker's agent*>. From Fig. 2 goal *number of units sold* at the line-of-sale grain, the retrieved trio for the iteration grain line-of-sale is <*Product ID*, *which*, *product*>. For each distinct trio found:
   - 2.1 If no dimension table has yet been created for the marker's agent:
     - 2.1.1 Create a dimension using the marker's agent name.
     - 2.1.2 Create a foreign key between the iteration's fact table and the new dimension.
   - 2.2 Creates a column in the marker's agent dimension, using the marker's agent name.

Fig. 4 shows a multidimensional model generated by using the fulfilled goals at Fig. 3 with the iteration grain line-of-sale. The picture is a partial screen capture from a template configuration tool devised by the authors of this paper.

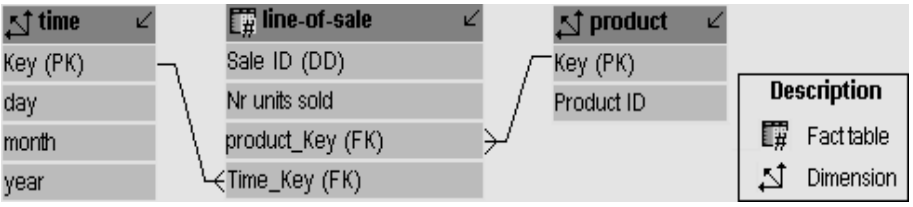


Fig. 4. Multidimensional model generated using the fulfilled goals at Fig. 3 and an iteration grain line-of-sale (*DD=Degenerated Dimension*; *FK=Foreign Key*; *PK=Primary Key*)

5 Conclusions

In this paper we have proposed the use of dimensional templates for automating the multidimensional design of DWs. Dimensional templates are built with a high level of abstraction, thus lowering their management complexity. This is achieved by the use of rationale diagrams, logical models that map end-user requirements to the types of data required to satisfy them.

We believe that our approach is particularly useful in DW prototyping environments, since (i) a dimensional template works as a pre-built solution and (ii) the configuration of templates to generate multidimensional models can be achieved without DW knowledge. These are key features regarding DW prototypes, since these systems highly benefit from a fast boot start plus low cost operations due to their experimental status.

Also, our approach suites better the purpose of automating the multidimensional design in DW prototypes than other existing proposals of the kind. These other automation methods require either extended periods of source data analysis by DW designers, DW design expertise or even exact source data documentation in specific formats: three requirements not compliant with the time and cost requirements of embryonic solutions such as DW prototypes. Even though our approach also requires DW expertise (to build dimensional templates), this initial effort is compensated by the re-usage capacity of the solution and by the lack of time-consuming interaction between DW experts/organizations' end-users that other approaches depend on.

Our approach is fully supported by two prototype tools developed by the authors: a *template builder tool* for creating and managing the rationale diagrams (used to generate Fig. 2); a *template configuration tool* for generating multidimensional designs from dimensional templates (used to generate Fig. 4) and also the related documentation in the *Common Warehouse Model* standard [21]. This last feature, not in the scope of this paper, enhances the scalability feature of the prototyped products, since many ETL tools can import the generated structures.

Interesting future work can be performed as a completion to the work presented in this paper. This includes the semi-automated creation of dimensional templates from real multidimensional designs.

References

1. Look Before You Leap,
`http://www.intelligententerprise.com/010216/feat3_1.jhtml`

2. Data Warehouse Prototyping: Reducing Risk, Securing Commitment and Improving Project Governance, http://www.wherescape.com/white-papers/white-papers.aspx
3. Huynh, T., Schiefer, J.: Prototyping Data Warehouse Systems. In: Kambayashi, Y., Winiwarter, W., Arikawa, M. (eds.) DaWaK 2001. LNCS, vol. 2114, pp. 195–207. Springer, Heidelberg (2001)
4. The Data Warehouse Budget, http://www.datawarehouse.inf.br/Papers/inmonbudget-1.pdf
5. Adelman, S., Dennis, S.: Capitalizing the DW (2005), http://www.dmreview.com/
6. Vrdoljak, B., Banek, M., Rizzi, S.: Designing Web Warehouses from XML Schemas. In: Kambayashi, Y., Mohania, M., Wöß, W. (eds.) DaWaK 2003. LNCS, vol. 2737, pp. 89–98. Springer, Heidelberg (2003)
7. Giorgini, P., Rizzi, S., Garzetti, M.: Goal-Oriented Requirement Analysis for Data Warehouse Design. In: DOLAP 2005, 8th International Workshop on Data Warehousing and OLAP, pp. 47–56. ACM Press, New York (2005)
8. Mazón, J., Trujillo, J.: A Model Driven Modernization Approach for Automatically Deriving Multidimensional Models in Data Warehouses. In: Parent, C., Schewe, K.-D., Storey, V.C., Thalheim, B. (eds.) ER 2007. LNCS, vol. 4801, pp. 56–71. Springer, Heidelberg (2007)
9. Song, I.Y., Khare, R., Bing, D.: SAMSTAR: A Semi-Automated Lexical Method for Generating Star Schemas from an Entity-Relationship Diagram. In: DOLAP 2007, 10th International Workshop on Data Warehousing and OLAP, pp. 9–16. ACM Press, New York (2007)
10. Abelló, A., Samos, J., Saltor, F.: YAM2 (Yet another multidimensional model): An extension of UML. In: International Symposium on Database Engineering & Applications. IEEE Computer Science, pp. 172–181. IEEE Computer Society, Washington (2002)
11. Luján-Mora, S., Trujillo, J., Song, I.Y.: Extending the UML for multidimensional modeling. In: Jézéquel, J.-M., Hussmann, H., Cook, S. (eds.) UML 2002. LNCS, vol. 2460, pp. 265–276. Springer, Heidelberg (2002)
12. Romero, O., Abelló, A.: Automating Multidimensional Design from Ontologies. In: 10th International Workshop on Data Warehousing and OLAP, pp. 1–8. ACM Press, New York (2007)
13. Alhajj, R.: Extracting the Extended Entity-Relationship Model From a Legacy Relational Database. Information Systems 28, 597–618 (2003)
14. Jensen, M., Holmgren, T., Pedersen, T.B.: Discovering Multidimensional Structure in Relational Data. In: Kambayashi, Y., Mohania, M., Wöß, W. (eds.) DaWaK 2004. LNCS, vol. 3181, pp. 138–148. Springer, Heidelberg (2004)
15. Kimball, R., Ross, M.: The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling, 2nd edn. John Wiley and Sons, Inc., USA (2002)
16. Malinowski, E., Zimányi, E.: Advanced Data Warehouse Design: From Conventional to Spatial and Temporal Applications. Springer Publishing Company, Heidelberg (2008)
17. Wherescape RED, http://www.wherescape.com/
18. Alkis Simitsis's list of ETL tools, http://www.dbnet.ece.ntua.gr/~asimi/ETLTools.htm
19. Mazón, J., Pardillo, J., Trujillo, J.: A Model-Driven Goal-Oriented Requirement Engineering Approach for Data Warehouses. In: RIGIM 2007, 1st International Workshop on Requirements, Intentions and Goals in Conceptual Modeling, pp. 255–264. Springer, Heidelberg (2007)
20. Kimball, R., Caserta, J.: The Data Warehouse ETL Toolkit: Practical Techniques for Extracting, Cleaning,Conforming, and Delivering Data. Wiley Publishing, Inc., USA (2004)
21. Vetterli, T., Vaduva, A., Staudt, M.: Metadata Standards for Data Warehousing: Open Information Model vs. Common Warehouse Metamodel. ACM SIGMOD Record 29, 68–75 (2000)

Multiview Components for User-Aware Web Services

Bouchra El Asri, Adil Kenzi, Mahmoud Nassar,
Abdelaziz Kriouile, and Abdelaziz Barrahmoune

SI2M, ENSIAS, BP 713 Agdal, Rabat, Morocco
elasri@ensias.ma, adil.kenzi@gmail.com
{nassar,krouile}@ensias.ma, barrahmoune@yahoo.fr

Abstract. Component based software (CBS) intends to meet the need of reusability and productivity. Web service technology leads to systems interoperability. This work addresses the development of CBS using web services technology. Undeniably, web service may interact with several types of service clients. The central problem is, therefore, how to handle the multidimensional aspect of service clients' needs and requirements. To tackle this problem, we propose the concept of multiview component as a first class modelling entity that allows the capture of the various needs of service clients by separating their functional concerns. In this paper, we propose a model driven approach for the development of user-aware web services on the basis of the multiview component concept. So, we describe how multiview component based PIM are transformed into two PSMs for the purpose of the automatic generation of both the user-aware web services description and implementation. We specify transformations as a collection of transformation rules implemented using ATL as a model transformation language.

Keywords: Information System Modelling, UML, View, Viewpoint, VUML, Multiview component, User-aware service, MDA, MVWSDL.

1 Introduction

With the popularity of the Internet and web-based access to information, software development must face up to heterogeneous environments and changing client's needs. In this context, reusability and interoperability are key criteria. Component based software (CBS) construction intends to meet the reusability need. The basic idea is to allow developers to reuse simple units of software called *components* to build up more complex applications. Web service is a technology that intends to meet the interoperability need. It addresses the requirement of loosely coupled, standard based and protocol independent distributed computing.

This work addresses the development of CBS using web services technology. Undeniably, web services are not dedicated service clients; rather they are exposed to a large public through the Internet. This is why web service providers try to develop and publish services which can be personalized to potential clients.

To develop such web services, the web service variability among various service clients must explicitly be analyzed and designed. However, current works usually

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 196–207, 2009.
© Springer-Verlag Berlin Heidelberg 2009

focus on defining processes for the development of web services without separating users' concerns. To tackle this problem, we propose the concept of multiview component as a first class modelling entity that allows the capture of the various needs of service clients by separating their functional concerns.

Certainly, some approaches have been put forward to take into account the variability of service clients' needs for instance by adapting their profiles [1] [2] or by managing their access rights [3] [4] or their contexts [5] [6]. Nevertheless, to the best of our knowledge, there is no approach that allows the modeling of users' needs by separating their concerns early in the lifecycle of the development. Thus, we propose in this paper, a process to develop user-aware web services tracking users concerns throughout the development lifecycle. To this end, we define the concept of the multiview component as a first class modeling entity which permits the representation of the needs and requirements of users by separating their concerns. The multiview component is a new modeling entity that provides, in addition to the simple interfaces, the multiview interfaces which have the characteristic of being flexible and adaptable to the different types of service clients. On the basis of the multiview component, we firstly elaborate the PIM which describes the structure and the functionalities of systems according to the different actors. Secondly, we define two transformations targeting two PSMs for the purpose of the automatic generation of both the multiview component description and resulting web services implementation.

The first transformation aims the generation of the multiview component service description. For this objective, we have defined a lightweight extension of the WSDL standard. It allows the representation of the component services' interfaces as well as the information about actors interacting with the component services.

The second transformation aims the generation of the Java code which constitutes the implementation of the resulting user-aware web services. For this end, we have defined a set of transformation rules targeting a J2EE Platform.

Finally, mapping to Platform Specific Model and code generation is done by specifying transformations as a collection of rules implemented in ATL.

The rest of this paper is structured as follows: Section 2 gives a brief overview of our motivating example. Section 3 describes the concept of multiview component. Section 4 presents our framework for developing user-aware web services. Section 5 presents some related works, and in Section 6 we give a conclusion and perspectives to our work.

2 User-Aware Web Services: A Running Scenario

In our study, we are guided by a motivating scenario which highlights our interest. It focuses on a set of course web services (WS), for the DLS system, published throughout the net. Those WS can be accessed by different users (students, professors, administrators, etc.) as by applications. It allows distant students to apply for courses, access to related documentation (slides, web pages, text, etc.), make exercises, communicate with teachers, and take exams. It allows professors to edit their own courses; plan learning experiences and units of work and record student assessments. The DLS allow the administrator to record students for available courses and manage human and material resources.

To highlight our interest, we consider the case of John and Alice who interact with the DLS. John is a student who is looking for subscribing at specific courses while Alice is a professor editing documentation for a Training course. Both John and Alice require the same web service. But, we want to offer them pertinent functionalities that exactly cope with their interest. So the service provider must adapt the component service access and behavior according to the current user profile. Thus, for John profile, the service provider must prepare lists of courses description (syllabus, price, schedule, etc.) and verify if there are any available places. So, John can choice the appropriate course, apply for subscription and pay, if even accepted, the course scholarship. The same service provider must prepare, for Alice, courses description (syllabus, schedule, material requirement) and verify if she is responsible for such courses. So, Alice can validate schedule, reserve materials, edit documentation or propose exams.

Implementing a Course service that realizes all functionalities required by John, Alice and others is not enough to offer configurable, manageable and reusable web services. To support different outcome, such service have to be user-aware one. It needs to meet different business domains and provide multiple service interfaces as response to each user's needs.

In order to meet this purpose, we propose the notion of multiview component as a new business concept which separates common business activities that copes to all business domains from those that are specific to a particular kind of users. The next section introduces and defines this concept.

3 The Multiview Component Model for User-Aware Web Services

In this section, we first give definitions related to component concepts and the view one. Then we describe some details about the structure of a multiview component and related mechanisms.

3.1 Preliminary 1: The Component Concept

C. Szyperski defines a component as a unit of composition with contractually specified interfaces and fully explicit context dependencies that can be deployed independently and is subject to third-party composition [7]. This definition is closed to that of B. Meyer who considers a component as an oriented client software unit [8]. In general, a component is a unit of program which comprises at least two parts: a specification part of its interfaces and behaviours, and an implementation part that carries out its services. An interface is a collection of operations that are used to specify a service of a component [9].

3.2 Preliminary 2: The View Concept

The view concept is largely used in several fields, as a mean of separation of concern, such as Database Management System [10], Workflow [11], Web Services [1], [4], [5], [6], etc. Generally, the separation of concerns [12] helps in writing software that is modularized by concern; modeling concerns and their relationships and extracting concerns that are tangled with others.

For our team, we use views as a means of both assuring functional separation of concern and managing access right. Our view based approach, called VUML (View based Unified Modeling Language), revolves around three key concepts: *actor*; *base* and *view* [13]. An actor is a logical or physical person who interacts with the system. A base is a core entity which includes specifications that are common to all types of actor. A view is a satellite which modularizes a classifier specification depending on an actor profile and constraints. A view zooms into the specific feature which interests an actor. It adjusts the classifier specification. It is a dynamic snapshot of the functional changes that occur in a classifier specification according to a certain type of actor.

3.3 The VUML Multiview Component Model

Based on the view concept and on the component one, we define the concept of multiview component (cf. figure 1) as a first class modeling entity that highlights the user needs and requirements early in the development lifecycle of the component based systems. The multiview component permits the capture of the various needs of component service clients by separating their functional concerns. For each component service client, the component service must provide the required capabilities that correspond to the needs of users invoking the component service. From an analysis/design viewpoint, the central problem therefore, is how to model the multidimensional aspect of the needs of the various actors interacting with the same component service. Thus, a multiview component provides in addition to the simple interfaces, the multiview interfaces (*MVInterface*) which are able to describe the capabilities of the component services according to the profiles of its requester.

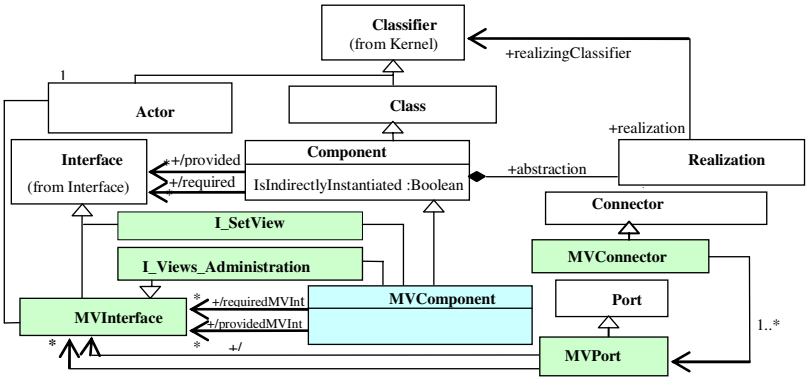


Fig. 1. Static structure of a multiview component

Figure 2 bellow illustrates an MVInterface provided by the Course MVComponent for the DLS case study. Such component is a multiview one since the outcomes of this component are in interaction with three actors: the professor, the student and the administrator. Each actor has specific needs in the *Course* component service. Thus, the *Course* component provides a multiview interface "Course". Such an interface is

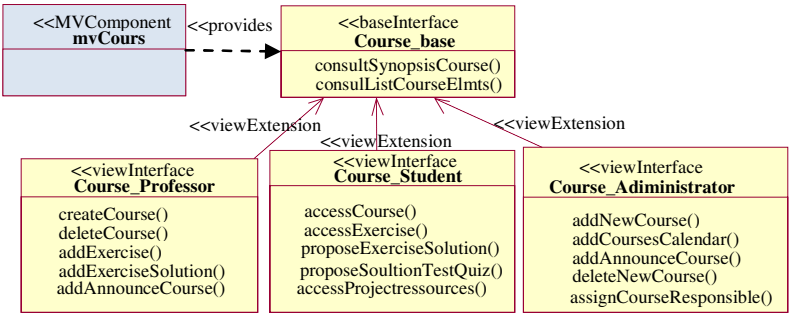


Fig. 2. The course multiview component

composed of a base interface (baseInterface) and a set of view interfaces (viewInterface). The baseInterface is a shared interface. It permits the representation of the functionalities of the component services required by all kinds of users. In contrast, the viewInterface permits the representation of the functionalities required by a specific kind of user. These functionalities are accessible only if the specific user is in interaction with the component service. The viewInterface depends on the base interface in the sense where the functionalities of the base interface are implicitly shared by all views interfaces.

Thus, the three actors can invoke the c*onsultSynopsisCourse( )*and *consultListCourse* included in the base interface "baseInterface" *course*. On the other hand, interfaces associated with each actor: The *Course_administrator*, *Course_Student* and *Course_Teacher* view interfaces are associated respectively to the administrator, student and teacher. They are accessible for the corresponding actor. More precisely, an Administrator can invoke only the operations belonging to *Course_Administrator* view interface while a student can only invoke the operation belonging to the *Course_Student* one.

4 From Multiview Component to User-Aware Web services: A Rule Based Approach

In order to take advantage of the MDA approach for the development of software systems, we define an approach for the development of user-aware web services governed by a set of transformation rules. Therefore, we firstly define a multiview component based PIM. This PIM reflects the structure and functionalities of the system according to actors interacting with each component service. Then we define two complementary transformations targeting two specific platforms. Each transformation is carried out on two steps: a model to model transformation and a model to code transformation.

The first transformation aims the generation of the multiview component description. For this reason, we have defined an extension of the standard of WSDL called Multiview WSDL (MVWSDL) that allows the description of the multiview component as will be illustrated in section 4.1.1. The second transformation aims the automatic generation of the code (Java) from the multiview component based PIM.

4.1 From the Multiview Component Based PIM to MVWSDL Code

The objective of this section is the description of the transformation of the multiview component based PIM into a MVWSDL description. First of all, we present the MVWSDL, our extension of the standard WSDL in order to describe the multview component and it's meta-model. Secondly, we present the transformation of the PIM into MWSDL based PSM by identifying the equivalence between source and target meta-model and the definition of the associated transformation rules.

4.1.1 The Multiview WSDL

WSDL is a W3C standard used to describe the interfaces of web services. In our approach, we propose MVWSDL (MultiView WSDL) as a lightweight extension of the standard WSDL. In fact, the WSDL standard defines an XML Schema for describing a service. This Schema is composed of six elements (Types, Messages, PortTypes, Binding, Port, Service) which allow to define service interfaces, theirs operations, the input/ouput parameters of each operation, the type of these parameters and the access point (URL) of the operations of services. However, this standard does not take into account the profile of users that interact with the service. Indeed, two service clients with different profiles can obtain the same WSDL.

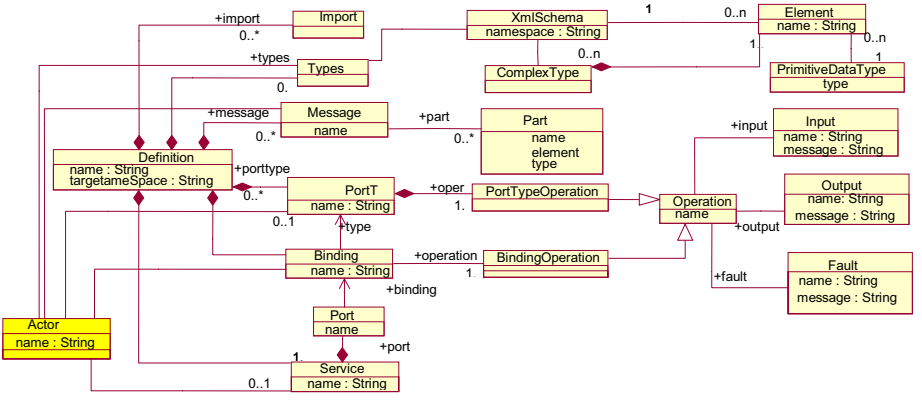


Fig. 3. The MVWSDL meta-model

To tackle this problem, we define an extension of the standards WSDL called MVWSDL (MultiView Web Service Description language) in order to describe multiview component. The objective of this extension is double. On one hand, it describes in a single XML file all component service interfaces both simples and multiview. On the other hand, it allows the adaptation of this description by providing a standard WSDL description tailored to the users' profiles. We have also established the Multiview WSDL Meta-model (figure 3) as an extension of WSDL meta-model. This extension is carried out by means of an element called "actor". Such an element permits the definition of the profile interacting with the component service. It is associated with the main elements of WSDL such as Types, Message, PortTypes, Binding and Service.

4.1.2 From the Multiview Component Based PIM to MVWSDL Based PSM

In the MDA approach, the transformation from a PIM into PSM firstly requires the specification of mapping between the source and the target meta-model. The specification of mapping consists on the definition of equivalences between meta-model source elements and meta-model target ones. Two or more elements of different meta-models are equivalent if they are compatible and they cannot contradict each other. In our approach, we have defined the multiview component meta-model and the MVWSDL meta-model respectively as source and target. Table 1 illustrates the equivalences between the two meta-model elements and the associated transformation rules:

After identification of the equivalences between meta-model elements, we proceed to the definition and the implementation of transformation rules by using the ATL language [15]. For illustration, we purpose the transformation *Viewinterface2WSDL.*

Table 1. Mapping from multiview component meta-model to the MVWSDL one

Elements of Multiview Component Metamodel	Elements of MVWSDL Metamodel	Transformation rule
MVComponent	Definition	MVComponent2definition
BaseInterface	Types, XMLSchema PortType, Binding, Service	BaseInterface2WSDL
viewInterface	Types, XMLSchema, Portype, Binding, Service	Viewinterface2WSDL
Parameter	Part	Param2part
Operation	PortypeOperation,BindingOperation, Input, Ouput, ComplexType	Operation2operation
dataType	Types	dataType2type

The rule *Viewinterface2WSDL* permits the creation of instances of five elements of the MVWSDL: PortType, Types, Binding, Port and Service. Each instance of element PortType is initialized with the attribute and references of the element viewInterface. Thus, the name is set according to the value v.name. The actor is initialized with the name of the actor (v.actor.name) associated to the *MVComponent!viewInterface.* The operations reference is assigned with all *MVWSDL!PortTypeOperation* created according to the *MVComponent!Operation* owned by the current *MVComponent!viewInterface.*

The rule *Viewinterface2WSDL* (cf. figure 4) create an instance of the element Types for each actor and its associated schema. For each generated schema, the namespace attribute is initialized with the name of the viewInterafce and its *complextype* is assigned with a collection of complex type generated by the rule Operation2opertaion. The rule *Viewinterface2WSDL* also creates an instance of *MVWSDL! Binding*. The name is set with the name of *v.name* + *'binding'*. The actor is initialized with the name of the actor associated to the *MVComponent!viewInterface*. The *porttype* reference is assigned with the variable *out*. The *boperations* is assigned with all *WSDL!BindingOperation* created according to the *MVComponent!Operation* owned by the current *MVComponent!viewInterface*. It also creates *MVWSDL! Service* and initializes it.

```
rule Viewinterface2WSDL{
  from v: MVComponent!viewInterface
   to out : MVWSDL!PortType(
     name <-v.name,
     actor <-v.actor.name,
     operations<- v.operations_v-> collect (
      x |thisModule.resolveTemp (x,'wsdlop'))),
 types :MVWSDL!Types(actor <- v.actor.name,xmlschema <-xschema),
 xschema :MVWSDL!XMLSchema (namespace<- v.name,
 complexType <- v.operations_v ->
 collect ( x|thisModule.resolveTemp(x,'complextype_IN'))->
   union(v.operations_v -> collect (
     x |thisModule.resolveTemp(x,'complextype_OUT')))),
 bd: MVWSDL!Binding(name <-v.name + 'Binding',…
```

Fig. 4. Excerpt from the ATL code of the viewinterface2WSDL transformation rule

4.1.3 MVWSDL Based PSM to MVWSDL Code

Transformation from the multiview component based PIM into MVWSDL based PSM allows the generation of a MVWSDL model that conforms to the MVWSDL meta-model.

```
<?xml version="1.0" encoding="UTF-8"?>
<definitions name="Course"  targetNamespace="urn://Course.wsdl">
<Types  actor = "allactors">…</Types>
<Types actor = "Teacher">...</Types>
<Types actor = "Student"> … </Types>
<message name="addExerciceRequest" actor="Teacher">...  </message>
<portType name="Course_base"  actor="allactors">
   <operation name="consultSynopsisCourse">
      <input name="consultSynopsisCourseRequest"/>
       <output name="consultSynopsisCourseResponse"/>
  </operation>…  </portType>
<portType name="Course_Teacher"  actor="Teacher">
   <operation name="addExercice">… </operation>...     </portType>
<portType name="Course_Student"  actor="Student">
    <operation name="postMessageForum">...</operation>....</portType>
</definitions>
```

Fig. 5. Excerpt from the MVWSDL generated code

This model is not the final implementation but has the necessary information to generate part or all the code. The generation of the code in ATL requires the definition of additional transformations on the basis of a set of helpers (functions) which are defined in the context of the MVWSDL meta-model elements. The MVWSDL generated code represents the definition of the interfaces of components according to the actors that will interact with the component. The code below presents the MVWSDL document of the *Course* multiview component.

The figure 5 illustrates an extract of the code generated according to the MVWSDL meta-model. In this extract, we only focus on the element PortType because the other elements are generated in the same manner.

4.2 From Multiview Component Model to Java Code

A web service has a description and an implementation. To generate automatically the mulltiview component description as well as the multiview component implementation, we have defined two transformations from the same PIM targeting several platforms. The first transformation allows the generation of the multiview component description discussed in the previous sections. The second transformation permits the generation of the multiview component implementation according to a particular implementation platform (dotNet, JWSDP, etc.). In our approach, we have chosen JAX-RPC as an implementation platform.

4.2.1 From Multiview Component Based PIM to PSM (JAX-RPC)

To automatically generate the multiview component implementation targeting a JAX-RPC implementation platform, we firstly have defined a JAX-RPC meta-model (cf. figure 6). Secondly, we specify the mapping of the multiview component meta-model as source meta-model to the JAX-RPC meta-model as a target meta-model by identifying the equivalent elements (cf table 2). Thirdly, we define the transformation rules which implement the equivalences between the source and target meta-model elements. Finally, we define additional transformations in order to generate the java code as an implementation of the multiview component.

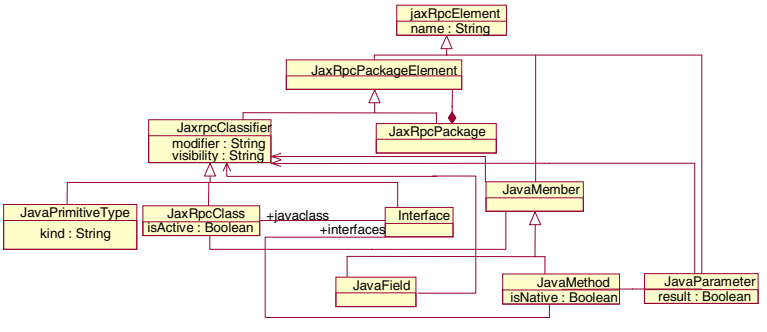


Fig. 6. JAX-RPC meta-model

Table 2. Mapping from multiview component meta-model to JAX-RPC

Elements of Multiview Component Metamodel	Element of JaxRPC Metamodel	Transformation rule
Package	JaxRpcPackage	Package2jaxrpcPackage
MVComponent	JaxrpcClass	MVCComponent2jaxrpcClas
BaseInterface	Interface	Binterface2Interface
ViewInterface	Interface	viewinterface2interface
Parameter	JavaParameter	Param2part
Operation	JavaMethod	Operation2Method
PrimiveDataType	JavaPrimitiveType	Data2Primitive

4.2.2 The Generated Code from the Multiview Component Based PIM

In JAX-RPC, a JaxrpClass which implement the functionalities of a service must implement java interface that extends java.rmi.remote and its methods throw java.rmi.RemoteException. Thus, we generate for each interface type of the multiview component (viewinterface, baseinterface) a java interface which extends the interface Remote. For each multiview component, we generate a class which implements all multiview component interfaces. Figure below depicts the code generated from the multiview component course. This multiview component provides the multiview interface which is composed of the baseInterface with the name "Course_base" and a set view interfaces corresponding to the actor "student" and "teacher". To illustrate our approach, we have chosen only two operations for each type of interface (cf. figure 7).

```
rule MVCComponent2jaxrpcClass {

package DLS;

public class Course implements  Course_Teacher,Course_Student,Course_base

{

public String addExercice(Integer exerciseID, String Exercise) {

        //[Implementation Code to beCompleted]}

public String addExerciseSolution(Integer ExerciseId,
```

Fig. 7. The generated class implementing the course mutiview component

5 Related Works

The view concept is largely used in variety software engineering domains such as workflow [11] or in object oriented approach [13]. In service oriented approaches, view concept is used in different approaches taking end-user into account. Thus, Maamar et al. use a view as a dynamic snapshot of the environmental changes that occur in a composite service's entire specification according to a certain context [5]. The difference between our definition of Views and the definition of the views by Maamar is the objective of using views. In our approach the view is used for the functional separation of concerns. In Maamar et al., the view is associated to a context (time, location, environmental changes).

Fink et al. propose a view based approach to manage access rights to operations in a single web service [4]. For this purpose, the authors define an access control model called VBAC (view Based Access Control) specifically designed to support the management of access control policies in distributed systems. In the VBAC model, views are used to group permissions or denials to access. This approach is similar to ours in the granularity level because in our approach, we treat the component service operations structured in interface.

Fuchs defines Service views to segment the WSDL of a set of web services according to specific kind of user analogous to views in DBMS [6]. This approach is too similar of ours; the main difference is granularity and abstraction level. Fuchs applies a view to a set of web services in contrast to our approach in which we apply a view to a single component service. Moreover, Fuchs uses a command language to generate a view of a set of web services, but in our work we use view to define modeling entity that encapsulates the user's needs and requirements.

To take end-user into account several approaches defining architecture, concepts and processes have been put forward. Thus, Chang et al., introduce UCSOA (User Centric Service Oriented Architecture) that allows end-users to compose applications [16]. It permits to service providers to discover the needs of end- users in order to match them. Tao et al. propose formal design approach for developing differentiated service, in which service outcomes depends on the profile of users interacting with the service [2]. The same service can provide differentiated functionalities corresponding to the needs of the service clients.

These last approaches introduce concepts, architecture and processes for adapting service to its end users that are closer to our approach. The specificity of our proposal is the modeling of end-users needs and requirements and the automatic generation code from high level models.

6 Conclusions

The adaptability and flexibility are challenging issues in the design of CBS. In this regard, we have presented in this paper, an effective framework for the development of highly adaptable and flexible web services. To this end, we have put forward the multiview component concept as a first class modeling entity which highlights the user's needs and requirements by separating their concerns early in the development lifecycle. Multiview component model reflects, then, the structure and the functionalities of system according to the actors which will interact with each service. Such model is transformed targeting two PSMs in order to automatically generate both the multiview component description and user-aware web service implementation. The multiview component description is generated according to the MVWSDL, our extension of the WSDL.

Once the MVWSDL description and the architectural java code are generated, it remains to generate WSDL document that are dynamically tailored for each actor and the functional code of the methods to have the complete system description and implementation. For these purposes, we are finalizing modules called "MVWSDLAdpater" to adapt WSDL documents. In the other hand we are working on the integration of the concept of viewpoint in dynamic modeling with UML (mainly activity and state-transition diagrams) to take into account this dynamic aspect in java code generation.

The merit of our model driven framework can be summarized in three folds: (i) the flowed process for developing user-aware web services; (ii) the separation of user concerns throughout the whole lifecycle; (iii) the advantage of considering the current standards of WSDL and java code generation.

References

1. Chang, S.H., Kim, S.D.: A Comprehensive Approach to Service Adaptation. In: IEEE International Conference on Services Oriented Computing and Applications SOCA (2007)
2. Tao, T.A., Yang, J.: Supporting Differentiated Services With Configurable Business Processes. In: Proc of the IEEE International Conference on Web Services (ICWS) (2007)
3. Alam, M., Seifert, J.P., Zhang, X.: A Model-Driven Framework for Trusted Computing Based Systems. In: EDOC 2007, pp. 75–86 (2007)
4. Fink, T., Koch, M., Oancea, C.: Specification and Enforcement of Access Control in Heterogeneous Distributed Applications. In: Jeckle, M. (LJ) Zhang, L.-J. (eds.) ICWS-Europe 2003. LNCS, vol. 2853, pp. 88–100. Springer, Heidelberg (2003)
5. Maamar, Z., Benslimane, D., Ghedira, C.: A View-based Approach for Tracking Composite Web Services. In: ECOWS 2005, pp. 170–181 (2005)
6. Fuchs, M.: Adapting Web Services in a Heterogeneous Environment. In: Proc of the IEEE International Conference on Web Services (ICWS 2004), pp. 656–664 (2004)
7. Szyperski, C.: Component Software - Beyond Object-Oriented Programming, 2nd edn. Addison-Wesley, Reading (2002)
8. Meyer, B.: What to compose. Software Development, mars, Online: Software development columns (2000), `http://www.sdmagazine.com/articles/2000/0003`
9. Kruchten, P.: Modelling Component Systems with the Unified Modelling Language. Rational Software Corp. (1999)
10. Rafanelli, M.: Multidimensional Databases: Problems and Solutions Idea Group (2003)
11. Chebbi, I., Dustdar, S., Tata, S.: The view-based approach to dynamic inter-organizational workflow cooperation. Data Knowl. Eng. 56(2), 139–173 (2006)
12. Ossher, H., Tarr, P.: Using multidimentional separation of concern to (re)shape evolving software. Communication of the ACM 44(10), 43–50 (2001)
13. Nassar, M., Coulette, B., Crégut, X., Ebsersold, S., Kriouile, A.: Towards a View based Unified Modeling Language. In: Proc. of 5th International Conference on Enterprise Information Systems (ICEIS 2003), Angers, France (2003)
14. El Asri, B., Nassar, M., Coulette, B., Kriouile, A.: Multiview Components for Information System Development. In: ICEIS 2005, pp. 217–225 (2005)
15. Atlas Transformation Language, `http://www.eclipse.org/m2m/atl`
16. Chang, M., He, J., Tsai, W.: T., Xiao, B., Chen, Y.: UCSOA: User-Centric Service-Oriented Architecture. IEEE International Conference on e-Business Engineering (ICEBE), pp. 248–255 (2006)

Knowledge Based Query Processing in Large Scale Virtual Organizations

Alexandra Pomares[1,2], Claudia Roncancio[1], José Abásolo[2], and María del Pilar Villamil[2]

[1] LIG, Grenoble University, Grenoble, France
firstname.lastname@imag.fr
[2] University of Los Andes, Bogotá, Colombia
jabasolo@uniandes.edu.co,mavillam@uniandes.edu.co

Abstract. [1]This work concerns query processing to support data sharing in large scale Virtual Organizations(VO). Characterization of VO's data sharing contexts reflects the coexistence of factors like sources overlapping, uncertain data location, and fuzzy copies in dynamic large scale environments that hinder query processing. Existing results on distributed query evaluation are useful for VOs, but there is no appropriate solution combining high semantic level and dynamic large scale environments required by VOs. This paper proposes a characterization of VOs data sources, called **Data Profile**, and a query processing strategy (called *QPro2e*) for large scale VOs with complex data profiles. QPro2e uses an evolving distributed knowledge base describing data sources roles w.r.t shared domain concepts. It allows the identification of logical data source clusters which improve query evaluation in presence of a very large number of data sources.

Keywords: Large scale query processing, Virtual organizations, Ontologies.

1 Introduction

A Virtual Organization, VO, is a set of autonomous collaborating organizations, called **VO Units**, working toward a common goal. It enables disparate groups to share competencies and resources [1] like data and computing resources. This type of organization has evolved to a national and world-wide magnitude [2,3]. They introduce complex characteristics related to their business processes and reflected in the shared data.

This work concerns query processing in VOs with a high number of autonomous data sources provided by independent, but logically related, VO units. Autonomy and logical relationship's coexistence generates uncontrolled situations of data source overlapping, data replication and dynamicity. Such characteristics have to be recognized to allow correct and high-performance query processing in large scale contexts. Current query processing strategies in heterogeneous data environments (e.g. multi-databases) do not fit well in this context since they do not recognize data source relationships and most of them require accessibility of all data sources. On the other hand, most results of query

[1] This research is supported by Ecos-Colciencias C06M02/C07M02 and Pontificia Universidad Javeriana.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 208–219, 2009.
© Springer-Verlag Berlin Heidelberg 2009

processing on large scale contexts (e.g. P2P systems) do not provide enough semantic integration to fit VOs requirements.

This paper proposes a strategy of query processing in VOs that improves scalability and quality of final response using a source selection process fed by a distributed knowledge base. The knowledge base materializes VO semantics and knowledge facts describing the role that data sources play with respect to domain concepts. The knowledge base evolves to integrate the information discovered by executing queries. This increases the precision of source selection during query evaluation.

The first part of this paper (Section 2) contributes to the definition of the notion of VO *data profile*. It is the characterization of the context of data sharing in VOs, created to recognize the aspects to be considered by a VO query processor. Section 3 discusses related works on heterogeneous and large scale contexts. Section 4 presents the *QPro2E* query processing strategy using knowledge based source selection. Section 5 proposes methods of knowledge capture. Section 6 analyzes the evolution principle of the strategy. Finally, Section 7 concludes this paper and introduces future work.

2 Virtual Organizations Data Profile

This paper focuses on large scale VOs using a federated model where preexisting data sources provided by different VO units are shared under predefined cooperation agreements [4]. Each VO unit keeps the control of its local data sources. This section introduces VO data profiles of Federated VOs. Such profiles include characteristics that have to be mastered to provide efficient data sharing solutions.

The motivational case of this analysis are Health VOs sharing data of patients. The nature of patient behavior and the organ specialization of medical discipline contribute to the distribution of patient data in autonomous information systems. Paradoxically, processes of medical record reconstruction, evaluation of treatments, as well as other event-based medicine projects require complete information. Health VOs have to deal with chaotic environments of distribution that must be controlled to support the execution of inter-organizational business processes.

We established VO's data profiles using our experience in two nation-wide health VOs (South America and Europe), and from literature reporting works around large scale VOs [5], [3], [6].

The aim of defining a data profile is to characterize possible scenarios of data sharing in VOs. We identified characteristics related to the type of queries, to data sources, and to the physical environment.

Query related characteristics

Common Domain Concepts. Data in a VO is about a relatively small number of shared concepts related to a domain (e.g. patient, client, tissue, gen). VO units provide portions of knowledge related to such concepts. In the following, shared concepts are called **Virtual Data Objects** (VDO).

Variable Quality Requirements. As a consequence of the large number of users in large scale VO, the type, scope, and query requirements can be different. They may differ on

source preference, required quality, response size, and time constraints, among others (e.g. a research group's requirements vs. emergency medical team's requirements).

Data Sources Related Characteristics

Intentional and Extensional Sources Overlap. The first is when two or more sources manage the same or similar (subsets) data schemas (attributes), whereas the second is about sources containing the same data related to a single VDO instance (client, product, patient, etc.). As expected, intentional overlap is considerably high in large scale VOs due to their work around similar subjects (e.g. patients). Contrary, our analysis shows that the percentage of data sources that contains data about a particular instance of a VDO is substantially lower (e.g. patient A is covered only by 1% of data sources).

Fuzzy Copies. Integration efforts of VO unit's subgroups and the nature of some processes, generate not well-defined duplication protocols inside the VO. There is not explicit knowledge about the whereabouts of the original or the copy of one element of a VDO. The same instance may be duplicated in more than one data source without following an explicit model of coherence. Different type of inaccurate copies can be detected like reduced copies, fragmented copies, and unreal copies (they refer to the same instance but, were created independently).We call this circumstance a Fuzzy Copy Phenomenon.

Uncertainty on Data Location. Knowledge in VO follows uncertain pattern distribution. This means that the distribution of one VDO instance can be completely different than the distribution of another instance (e.g. patient1 data can be in data source a, b, and c, and patient2 data can be in data sources b, e, and f).

High Data Volume. The volume of data managed in national or global-wide VO is high in terms of the number of data sources and/or in terms of data contained in each one of them. Typically, the number of data providers is in the tens or hundreds and the volume that each one maintains may exceed the order of gigabytes.

Heterogeneous Data Sources. VOs have to deal with structural, syntactical and semantic heterogeneity.

Different Levels of Security Constraints. Data provided by each VO unit can have different security constraints. Large scale VOs may involve different policies for each VO data source.

Physical Environment Characteristics

Dynamic Environment. Although established levels of service between the VO units can exist, the autonomy of organizations added to the dynamic nature of sharing relationships in VOs [1] can give raise to unstable states in terms of available data sources, networks latencies, response rates, and so on.

Wide Area Network Distribution. Virtual" in VOs is related to the geographic distance between its participants. Distribution of VO units is mainly supported by WAN networks.

Some of the characteristics in VOs are considered by existing multi-database systems. Nevertheless, VO query processing requires a non-obvious integration process

affected by extensional overlapping, fuzzy copies phenomenon, and uncertainty of data location in a physical environment where it is not possible to assume stable conditions.

3 Related Work

The evolution in distributed query processing was driven mainly by the integration of enterprise databases, and the support of Internet communities. VOs are a recent intersection of these two worlds that profit from the experience of both.

3.1 Distributed Enterprise Databases

Generally speaking, technical advances in distributed enterprise databases[2] contexts can be categorized in *mediation systems* and distributed DBMS.

Mediation systems were born to provide applications with access to multiple heterogeneous and distributed data sources. Reference architecture of mediation systems [7] distinguishes the data source level, the wrapper level and the mediation level. Numerous efforts have been focused on assisting the creation of mediation systems (as [8,7,9] or on its evolution and adaptability as [10,11] by using ontologies to describe wrapping and mediation rules. Principles of mediation, like abstraction levels and wrapping, are profitable in VOs data sharing architectures. Nevertheless, mediation strategies do not focus on supporting a large number of independent but related data sources and do not scale well when there are several forms of overlapping, as in VOs.

A distributed DBMS manages databases partitioned and distributed in a controlled way between a group of dispersed nodes. Several results on query optimization, like shipping [12], semijoins [13], query parallelization [14], and adaptable physical operators [15] can be adapted to VOs. Contrary to VOs, distributed DBMSs optimize query execution based on known horizontal (data) and vertical (attributes) fragmentation. Additionally, they suppose stability of the network, and were not created to support more than tens of data sources.

3.2 Internet Communities

Managing distributed data resources is the nature of the Internet. Given the compatibility of VOs needs, this section focuses on projects to support virtual communities over wide scale infrastructures like peer to peer (P2P) networks.

P2P data management systems (PDMS) have exploded in the last three years. A PDMS is a new class of decentralized data sharing system that preserves semantics and rich query languages [16]. It consists of a set of autonomous peers that share (semi-)structured data sources and can participate in a global query evaluation. The most remarkable proposals of PDMS are PIER [17], PinS [18], SAINTEtiq [19], PIAZZA[20] and EDUTELLA [21]. PIER and PinS use a structured P2P network based on DHT (Distributed Hash Tables) and the others a Super Peer network. Their most important concern is scalability of query processing in networks of thousands of peers. PIER and PINS, for instance, use a hash function to efficiently distribute and retrieve data or metadata, respectively, in order to improve the selection of nodes containing relevant data

[2] Developed since more than 30 years!

for a query. PIAZZA, Edutella and SAINTetiq P2P use summaries of nodes contents in a super peer backbone that allows scalable query routing. EDUTELLA and SAINTetiq also include an interesting concept of a semantic cluster of nodes to reduce the scope of query location.

These projects represent important advances in data management in large scale contexts. Nevertheless, they are mainly designed for contexts where the amount of data in each source is not too high, like PIER, or where contents can be well-described using metadata indexes like in PINS. Semantic groups of participants proposed by EDUTELLA and SAINTEtiq seem to be a good idea in VOs. However, it is necessary to identify different domains between participants; that is not common in VOs. SAINTEtiq P2P proposes a linguistic summarization of DBs that appears to be very appropriate to optimizing query planning and execution in large scale contexts although, the evolution and quality of summaries are not already well-treated to support changing and complex schemas of databases.

In general, PDMS projects have focused on the optimization of the location process. This will be necessary for large scale VO. However, the integration process in PDMS is interpreted as the union of answers relevant to a query. This is not enough for VOs where more semantics are required. As illustrated in Section 2, in VOs the logical relation among data sources requires an important effort to integrate answers at the instance level. So, the challenge is to provide this in contexts involving numerous data sources.

4 Semantic Query Processing

VOs Data profile (Section 2) contrasted with current solutions in distributed contexts (Section 3) led us to identify that available strategies of query processing are not enough when complex factors like sources overlapping, uncertainty of data location, and fuzzy copies, coexist in dynamic large scale environments. This gap was the motivation to propose QPro2E (Query Processing based on Extensional Evolution), a strategy of query processing in VOs.

QPro2E is based on a mediation architecture improved to support complex and large scale VOs Data Profiles. The mediation level includes an evolving knowledge base (KB) that extends the domain description with metadata that clarifies the VO data profile. This extension is used to reduce the uncertainty during query processing. The evolution of the KB increases the precision of query processing decisions.

This section details QPro2E strategy. Section 4.1 presents the representation of the data profile as an ontology knowledge base. Then Section 4.2 presents the main decisions of the query processor QPro2E.

4.1 Ontology Knowledge Base

The KB is represented as an ontology in OWL [22]. It includes three initial classes: VOUnit(participants), VOResource and VODomainConcepts. During the evolution of the VO, these classes are specialized and related to better represent the data profile. Figure 1 illustrates a portion of the initial KB for the Health VO.

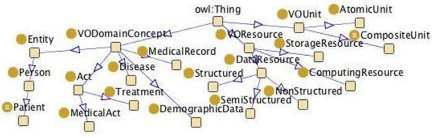


Fig. 1. VO Initial Knowledge Base

```
SELECT   ?treatmentDesc
WHERE{?Patient :hasDemographicData ?Demographic.
?Demographic :gender ?gender .
?Demographic :age ?age .
?Patient :hasMedicalRecord ?MedicalRecord .
?MedicalRecord  :hasMedicalAct ?MedicalAct .
?MedicalAct  :diagnosis ?diagnosis .
?MedicalAct  :hasTreatment ?treatment.
?treatment :treatmentDescription ?treatmentDesc
FILTER (?age < 14) .
FILTER regex(?diagnosis, "Cancer", "i"). }
```

Fig. 2. Query in SPARQL

VOUnits are the participants of the VO. A VO Unit can be atomic or composite. In the later case, it represents a temporal or a permanent group of atomic VO units working together around a specific collaboration process.
VOResources are physical or logical resources provided by VO units. DataSource, ComputingResource and StorageResource are three specializations of this class.
VODomainConcept includes the subclasses that describe the domain of the VO. For the health VO, initial classes are Patient, Disease, Medical Act, and so on.

Queries. As in traditional mediation systems, a VO query is a query over subclasses of the VO DomainConcept. However, when a class is recognized as a relevant class for a group of users, it is designated as a **Virtual Data Object**. This designation implies that queries over this class and its descendants will be optimized on the system. A query example in natural language is: Select **treatment** from VDO **Patient** where **age** $\leq$ **14** and **diagnostic = cancer**. Expressed in SPARQL [23] over the ontology of Figure 1 is presented in Figure 2.

Since VODomainConcept subclasses do not contain individuals, the execution of the query cannot be made directly over the ontology. Query evaluation requires knowledge of the data profile characteristics that can be useful to guide queries only to relevant data sources. This is explained in the following:

Data Profile Related Knowledge Facts. The lack of knowledge about data profile characteristics affects the efficiency of the query evaluation process. In order to reduce this problem, the KB includes a group of metadata expressed as knowledge facts that relate VODataSources with VDOs, and describe composite VOUnits. Their intention is to clarify the fuzzy copy phenomenon, the existence of data sources overlapping, and reduce the uncertainty of data location around the VDOs.

1. VODataSources - VDO: This type of fact describes wheter or not a data source plays a role w.r.t a VDO. Roles can be intentional or extensional. Intentional role means that a data source can resolve one (or more) property of the VDO. Extensional role means that a data source contains instances of the VDO when one or more of its properties are restricted (e.g. ChildPatient: Patient with age$\leq$14). This section defines the possible type of knowledge facts; Section 5 suggests how to obtain them. Possible intentional roles of data sources w.r.t. VDO are:

- knows: Its schema can be mapped to the schema of the VDO in one or more of its properties (e.g. HospitalDatabase knows Patient on properties DemographicData and EmergencyAct).

- primarySource: It is the original source of a group of properties of the VDO (e.g. MedicalLaboratoryDatabase is a primarySource for the property MedicalAnalysis). Possible extensional roles are:

- authority: It contains ALL the instances of the VDO when a particular restriction is applied (e.g. RegionalDB1 is an authority of Patient when the property region=region1).

- specialist: It contains PRIMARILY instances of the VDO when a particular restriction is applied (e.g. CardioInfantilDB is a specialist Patient when the property age $\leq$ 14).

- container: It contains at least one individual of the VDO when a particular restriction is applied (e.g. HospitalDB is a container Patient when the property diagnostic = Diabetes because it contains at least one individual with diabetes).

2. VOUnits - VOUnits: The relationship between VO units reflects business-to-business collaboration processes of participants inside the VO. These processes concretize the general objective of the VO and their logic provides a deep knowledge of the VO. A collaboration process between VO units is represented as a **Composite VO Unit**. This type of unit has as components the atomic VO units that participate in the collaboration process and the collaboration process itself. Uncertainty around extensional overlapping can be reduced through the recognition of existing Composite VOUnits. The hypothesis is that VO units that work together maintain in their data sources similar group of VDO instances, making them most susceptible to have extensional overlapping and fuzzy copies of VDOs.

4.2 Query Execution Using VO's Knowledge Base

In order to avoid the contact of unnecessary data sources, decrease the waiting for query answers and adapt automatically when some data sources are unavailable, the strategy of query processing overcomes VO data profile complexity using the available knowledge facts. The goal is to guide queries only to relevant data sources and evaluate the query cooperatively. It is composed of three tasks: query cartography creation, temporal data source clusters definition, and shared query execution.

1. Query Cartography Creation: Given a query Q, its query cartography designates the set of data sources relevant for the query with its associated role. A data source is relevant for a query if it plays an extensional or intentional role w.r.t. Q. To identify relevant data sources this task executes a query over the KB. Since knowledge facts are related to VDOs and not to queries, obtaining relevant facts is not an obvious process. The purpose is to identify the specializations of VDO nearest to the query restrictions and take their knowledge facts of the type VODataResource-VDO. Due to space limitations we only present the principle of the algorithm by example considering the query presented in Section 4.1. To find relevant knowledge (facts), the query is decomposed as follows:

* Main Restriction: **age** $\leq$ **14** and **diagnostic=cancer**
* Level 1 SubRestriction 1: **diagnostic=cancer** * Level 1 SubRestriction 2: **age** $\leq$ **14**
* Required Property 1: **treatment**
* Restriction Required Property 1: **diagnostic** * Restriction Required Property 2: **age**

Table 1. Query Cartography

Concept	Intentional Knowledge	Extensional Knowledge
treatment	DB1,DB2,DB7-knows	
age $\leq$ 14	DB2,DB4,DB9-knows	DB4-Container
diagnostic= cancer	DB5,DB6,DB8-knows, DB4-primSource	DB5-Authority, DB4-Specialist, DB6-Container

To obtain extensional knowledge the algorithm sends queries to the KB to obtain relevant facts for each restriction. The ideal case is to obtain facts related to the Main Restriction. If it is not the case it sends queries related to SubRestrictions. Finally, the algorithm asks all the intentional facts related to the RequiredProperty and to the Restrictions Properties. Although initial queries over a VDO will not have enough extensional knowledge facts associated, the evolution of the VO assures more availability of this type of knowledge. Table 1 illustrates the result for the example query.

2. Temporal Data Source Clusters Definition: Query cartography contains data sources relevant to resolve the query. Nevertheless, if one data source cannot resolve all the restrictions or does not contain all the properties, it is necessary to cooperate between data sources to resolve the complete query. This task creates clusters of data sources that will be in charge of executing the query and producing a subset for the final answer. The logic to define data source clusters is supported by the query cartography and the available composite VOUnits in the KB. It matches query cartography data sources against data sources that belong to the same composite VOunit, and creates a list of potential clusters. In order to assure that each cluster can efficiently execute the complete query, it does a refinement process over the initial clusters. Given the previous query cartography and the following composite VO Units:

- Public Hospitals: Unit1(DB1,DB6), Unit2(DB4, DB9), Unit3(DB2)
- RegionalHealthCareProviders: Unit1(DB1,DB6), Unit4(DB10, DB11), Unit5(DB7)
- DiseaseAssociations: Unit6(DB5), Unit7(DB3).

Potential clusters for Q are:

- Cluster 1: DB1, DB6, DB4, DB9, DB2 **- Cluster 2**: DB1, DB6, DB7 **- Cluster 3:** DB5

After the refinement process final clusters are:

- Cluster 1: DB1, DB6, DB4, DB9, DB2, DB5 **- Cluster 2**: DB1, DB6, DB7, DB4, DB5

DB5 was added to the first two clusters due to its interesting role of authority. The last cluster was deleted due to its incompleteness.

3. Shared Query Execution: Inside each cluster the query execution follows a collaboration model based on LINDA [24] tuplespace. Each data source is represented by a process whose priority and behavior depends on the role(s) its data source has w.r.t the query. Process communication is made indirectly using the set of primitives to store and retrieve tuples from the cluster tuplespace.

The interaction with the tuplespace is determined by the type of process. In priority order, a process may be: a (1) **Selector Process**, if it represents a data source that contains the most selective restriction of the query, (2) **Authority Process**, if it represents an authority data source, (3) **Specialist Process**, if it represents a specialist data

source, (4) **Regular Process** if it represents a data source with the role Container or Knows w.r.t. one of the query restrictions (5) **FinalPrimary Process** if it represents a data source with the role PrimarySource, (6) **FinalRegular Process** if it represents a data source with the role Knows w.r.t. one of the query properties. If a process represents a data source that plays more than one role its behavior takes into account all the roles. Contrary, its priority is determined by the most important role. There is also a **Supervisor Process** that may be independent of any data source. Each cooperative set must have only one supervisor and at least one selector process.

The logic of the execution is the following: The selector(s) process(es) puts into the tuplespace one tuple for each VDO instance that has been validated by its data source. Authority, Specialist, and Regular Processes, following their priority, take tuples which are not already evaluated on the restriction(s) that their data sources can validate, and were not already evaluated by them. If the VDO instance passes the evaluation, the process changes the tuple to indicate the restriction is already evaluated. If the evaluation fails by omission, if the process is Authority, the tuple is deleted. In the other cases, the tuple is maintained adding to the tuple the identification of the data source as an already evaluator. If the evaluation fails by negation, the tuple is deleted.

FinalPrimary Process and FinalRegular Process, following their priority, take tuples that are already evaluated on all the restrictions and returns the values of the properties they are able to provide. If a user specifies he wants to obtain only original data, FinalRegular Processes are omited during this step.

The supervisor process takes periodically complete tuples, gives them to the final user, and copies them to an answer tuplespace. If there are tuples that cannot be evaluated completely inside a cluster, the supervisor puts them into another cluster.

A query finishes when: (1) there are not tuples on the tuplespace of any cluster or (2) the user has received a number k of VDO instances (k was defined by the user) or (3) a system query timeout expires.

This model of execution assures a natural adaptability to unstable states on physical resources. A low response rate of processes is supported through the asynchronous communication in tuplespaces. Additionally, since each cluster can supply a property or a restriction by different processes, there is an inherent adaptability of data source failures during query execution. Similarly, the exchange of tuples between clusters assures, simultaneously, the adaptability to failures, and the solution of the problem of clusters that do not have extensional overlap.

5 Knowledge Facts Obtention

Data Sources roles and composite VOUnits can be acquired using three approaches: (1) Manually (e.g. expert's or DBA's definition of knowledge), (2) Interpreting the execution of processes, (3) Automatically extracting it from sources of knowledge. This section presents strategies for the last two approaches.

1. Query Process Interpretation: Considers a query Q submitted to the VO and assuming there is no knowledge (facts) related to Q restrictions, the query cartography creation will obtain only data sources with intentional roles. The execution of the query inside clusters will obtain instances that match Q associated with the group of

participant data sources. The principle of this strategy is to interpret the query answers as a group of knowledge facts. It identifies which data sources were able to resolve restrictions, and analyses the behavior of cooperative cluster exchanges. After this, it is possible to obtain new facts that relate data sources with the container role, and new artificial" composite VO units, respectively. An initial execution with interesting queries (most common restrictions) allows an initial feeding of the knowledge base.

2. Collaboration Process Interpretation: Business processes related to composite VO units are sources of knowledge of extensional overlapping and primary source role. The interpretation of a process description lets us identify the units and resources related to an activity, and the flow of elements between activities. If an activity creates an element and sends it to another activity (both related to different units) implies that units maintain the same group of instances (extensional overlap), this is used to create composite VOUnits. Similarly, data sources of units related to activities that create data are related using the role PrimarySource.

3. Profile Distinction: All sources are potentially specialists on a VDO. However, to evaluate each one of them is neither practical, neither viable. The proposal is to distinguish data sources with a trend to be specialists and to focus on them for knowledge extraction (data mining).

The reduction of the initial group of data sources is accomplished using the behavior of data sources during past query evaluations. Taking an interesting VDO property, it is evaluated to discover what data sources cooperated to resolve its restrictions. For interesting VDO properties the system identifies the data sources that cooperate to resolve its restrictions. Sources that have answered indistinctly different values restrictions are separated from those that have answered only a unique or a reduced group of values restrictions. The later case corresponds to the interesting data sources over which a data mining task can be applied to identify clusters of VDOs.

6 QPro2E Analysis

The main principle of QPro2E is its capability to evolve. This section presents the results of QPro2E validation from this point of view. The objective was to measure the impact on the number of contacted data sources during query processing accordingly to the available knowledge during query evaluation. We considered facts obtained from the query process interpretation.

We produced two groups of queries around the same VDO. Each query has three restrictions. The first group contains queries with restrictions on the same group of properties with compatible filter values. The second group was randomly created. We considered two data profiles. The first one has 100 data sources with an intentional overlap average of 30% and average extensional distribution of instances of 4%. The second one varies the extensional distribution to 10%.

Results in terms of percentage of useless data sources w.r.t the total number of contacted data sources are presented in Table 2.

As expected, the results are better for the first group of queries. Nevertheless, we observed how the difference in extensional distribution substantially decreases the percentage of useless data sources in both cases. This is due to the exigency of contacting

Table 2. Useless Data Sources Reduction

% of useless data sources	Data Profile 1		Data Profile 2	
Query	Group 1	Group2	Group 1	Group 2
First	57%	64%	43%	59%
Middle	25%	26%	11%	21%
Last	12%	14%	0%	11%

more sources to evaluate the instances that are more distributed. In general terms, this analysis shows the strong source reduction obtained by QPro2E. Nevertheless, QPro2E was designed for large scale contexts, and its application in contexts with a low number of data sources or without intentional overlapping is not recommended.

7 Conclusions

The starting point of this paper is the analysis of data sharing contexts in VOs. We illustrated their data profile to elucidate the characteristics that difficult query processing using existing solutions in distributed and heterogeneous systems. Characteristics as intentional and extensional overlap, fuzzy copies, uncertainty on data location coexisting inside highly distributed dynamic environments require novel strategies to assure scalability and quality of response.

This paper presents QPro2E, a query processing strategy for large scale VOs. It uses an evolutive and distributed knowledge base to improve data source selection. Such knowledge allows us to address queries exclusively to data sources that play appropiate roles w.r.t query requirements. Q2Pro2E logic includes a data integration process at instance level that allows the use of complementary data sources to provide high-quality query responses. It involves a simple but scalable query execution approach based on the tuple spaces coordination model to deal with dynamic an unstable environments.

Q2Pro2E is part of a large scale mediation system deployed on a grid infrastructure called ARIBEC [25]. It was created as an extension of the ORS strategy [26] when there is not a natural referential data source. Short-term work involves the global integration and test of our proposals in a large scale environment and more research on automatic discovery of extensional knowledge.

References

1. Foster, I., Kesselman, C., Tuecke, S.: The anatomy of the grid: Enabling scalable virtual organizations. Int. J. High Perform. Comput. Appl. 15, 200–222 (2001)
2. NEESGrid: Nees consortium (2008), http://neesgrid.ncsa.uiuc.edu/
3. BIRN: Bioinformatics research network - birn project (2008), http://www.loni.ucla.edu/birn/
4. Venugopal, S., Buyya, R., Ramamohanarao, K.: A taxonomy of data grids for distributed data sharing, management, and processing. ACM Comput. Surv. 38, 3 (2006)
5. Grethe, J.: et al: Building a national collaboratory to hasten the derivation of new understanding and treatment of disease. Studies in health technology and informatics 112, 100–109 (2005)

6. Chu, X., et al.: A service-oriented grid environment for integration of distributed kidney models and resources. In: Concurrency and Computation:Practice and Experience. Wiley Press, New York (2007)
7. Roth, M., Schwarz, P.: A wrapper architecture for legacy data sources. In: VLDB 1997, pp. 266–275. Morgan Kaufman, San Francisco (1997)
8. Garcia-Molina, H., Papakonstantinou, Y., Quass, D., Rajaraman, A., Sagiv, Y., Ullman, J.D., Vassalos, V., Widom, J.: The tsimmis approach to mediation: Data models and languages. Journal of Intelligent Information Systems 8, 117–132 (1997)
9. Doan, A., Domingos, P., Halevy, A.Y.: Reconciling schemas of disparate data sources: a machine-learning approach. In: SIGMOD 2001, pp. 509–520. ACM Press, New York (2001)
10. Melnik, S., Garcia-Molina, H., Paepcke, A.: A mediation infrastructure for digital library services. In: DL 2000: Proceedings of the fifth ACM conference on Digital libraries, pp. 123–132. ACM Press, New York (2000)
11. Bruno, G., Collet, C., Vargas-Solar, G.: Configuring intelligent mediators using ontologies. 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. 554–572. Springer, Heidelberg (2006)
12. Kossmann, D.: The state of the art in distributed query processing. ACM Comput. Surv. 32, 422–469 (2000)
13. Chen, M.S., Yu, P.S.: Combining joint and semi-join operations for distributed query processing. IEEE Trans. on Knowl. and Data Eng. 5, 534–542 (1993)
14. Wenlong, H., Xiaolin, L., Jixiang, J., Yu, F., Yi, X.: Data model and virtual database engine for grid environment. In: GCC 2007, pp. 823–829. IEEE Computer Society, Washington (2007)
15. Rundensteiner, E.A., Ding, L., Sutherland, T., Zhu, Y., Pielech, B., Mehta, N.: Cape: continuous query engine with heterogeneous-grained adaptivity. In: VLDB 2004, pp. 1353–1356 (2004)
16. Halevy, Y., Ives, G., Suciu, D., Tatarinov, I.: Schema mediation for large-scale semantic data sharing. The VLDB Journal 14, 68–83 (2005)
17. Huebsch, R., Hellerstein, J.M., Lanham, N., Loo, B.T., Shenker, S., Stoica, I.: Querying the internet with pier. In: VLDB 2003, VLDB Endowment, pp. 321–332 (2003)
18. Villamil, M., Roncancio, C., Labb, C.: Range Queries in Massively Distributed Data. In: Proc. Int'l WS on Grid and Peer-to-Peer Computing Impacts on Large Scale Heterogeneous Distributed Database Systems, Krakow, Poland (2006)
19. Hayek, R., Raschia, G., Valduriez, P., Mouaddib, N.: Summary management in p2p systems. In: EDBT, pp. 16–25 (2008)
20. Tatarinov, I., Ives, Z., Madhavan, J., Halevy, A., Suciu, D., Dalvi, N., Dong, X.L., Kadiyska, Y., Miklau, G., Mork, P.: The piazza peer data management project. SIGMOD Rec. 32, 47–52 (2003)
21. Nejdl, W., Wolf, B., Qu, C., Decker, S., Sintek, M., Naeve, A., Nilsson, M., Palmér, M., Risch, T.: Edutella: a p2p networking infrastructure based on rdf. In: WWW 2002, pp. 604–615. ACM, New York (2002)
22. Horrocks, I.: Owl: A description logic based ontology language. In: van Beek, P. (ed.) CP 2005. LNCS, vol. 3709, pp. 5–8. Springer, Heidelberg (2005)
23. Eric Prud, A.S.: Sparql query language for rdf (2007), http://www.w3.org/tr/rdf-sparql-query/
24. Carriero, N., Gelernter, D.: Linda in context. Commun. ACM 32, 444–458 (1989)
25. Pomares, A., Roncancio, C., Abásolo, J., Villamil, M.D.P.: Dynamic source selection in large scale mediation systems. In: Hameurlain, A. (ed.) Globe 2008. LNCS, vol. 5187, pp. 58–69. Springer, Heidelberg (2008)
26. Pomares, A., Abasolo, J., Roncancio, C.: Virtual objects in large scale health information systems. In: Studies in Health Technology and Informatics, pp. 80–89. IOS Press, Amsterdam (2008)

Applying Recommendation Technology in OLAP Systems

Houssem Jerbi, Franck Ravat, Olivier Teste, and Gilles Zurfluh

IRIT, Institut de Recherche en Informatique de Toulouse
118 route de Narbonne, F-31062 Toulouse, France
{jerbi,ravat,teste,zurfluh}@irit.fr

Abstract. OLAP systems offering multidimensional and large information space cannot solely rely on standard navigation but need to apply recommendations to make the analysis process easy and to help users quickly find relevant data for decision-making. In this paper, we propose a recommendation methodology for assisting the user during his decision-support analysis. The system helps the user in querying multidimensional data and exposes him to the most interesting patterns, *i.e*. it provides to the user anticipatory as well as alternative decision-support data. We provide a preference-based approach to apply such methodology.

Keywords: Decision-support analysis, OLAP, Recommendations, Preferences.

1 Introduction

OLAP (On-Line Analytical Processing) systems are the predominant frontend tools for decision-support systems. They provide a multidimensional view of the data as this is certainly the most logical way to analyze businesses and organizations. Data are organised according to subjects of analysis, called facts, which are associated to axes of analysis, called dimensions. A decision-support analysis is an interactive exploration of Multidimensional DataBases (MDB), which allows users to see data from different perspectives.

1.1 Context and Motivations

Decision-support systems intend to help knowledge workers (executives, managers, etc.) make strategic business decisions. As enterprises face competitive pressure to increase the speed of decision making, the decision-support systems must evolve to support new initiatives, such as providing a more personalized information access and helping users quickly find relevant data. Recommender systems are one way to meet this need. Recommender systems are best known for their use on e-commerce Web sites (Amazon [15], MovieLens [17]), where they use input about a customer's interests to provide advice on movies, travels, and leisure activities.

OLAP provides an interactive analysis of multidimensional data based on a set of navigational operations. In most cases, the analyst is expected to use these operations

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 220–233, 2009.
© Springer-Verlag Berlin Heidelberg 2009

intuitively to find interesting patterns [6,19]. Obviously, analysis process becomes very laborious and complex task due to the large size and the high dimensionality of OLAP data [5]. We argue that the manual effort and the time spent in analysis could be reduced by anticipating the user's strategy and recommending relevant data for decision-making. Furthermore, OLAP process brings the user into a world of endless possibilities when applied to a high dimensional and hierarchical dataset. Analysts are frequently confronted with several adjoining patterns of multidimensional data with various perspectives and different granularity levels, *e.g*. analysis of sales amounts by customer may be performed according to cities, zones, departments, regions, and states. Providing advice on relevant patterns reduces effectively the user faltering when exploring multidimensional data.

To meet the challenges of more user-centered decision-support systems, OLAP tools are to be extended with recommendation techniques to make the analysis process easy.

1.2 Related Work

Recommendation approaches have been studied in many research communities, such as information retrieval [2], World Wide Web [3], and databases [13,20]. Existing recommendation approaches are usually classified into the following categories, based on how recommendations are generated:

- content-based methods [14,16] recommend to the user items similar to the ones the user preferred in the past,
- collaborative Filtering [12,20] recommends to the user items that people with similar preferences liked in the past, and
- hybrid approaches that combine collaborative and content-based methods [3].

In OLAP, Giacometti et al [7] propose to recommend to the user the next query based on the OLAP server query log. Recommendations are provided irrespective of user preferences while such preferences play an important role in the success of recommender systems [3]. Besides, this approach consists in recommending full queries and does not consider flexible recommendations that deal with different levels of user involvement.

Recommender systems always assume that the target of the recommendations is the current user. Therefore, user modeling plays the main role in the success of these systems [3]. User modelling in OLAP has been studied in two main works. In [10], a context-aware preference model is proposed. This model deals with user interests that vary according to different contexts of OLAP analysis. Sapia [19] proposes to model user behavior in order to improve caching algorithms of OLAP systems. This approach deploys information about characteristic patterns in the user's data access.

1.3 Aims and Contributions

In order to make the analysis process easy, we intend to define a recommendation methodology for assisting the analyst. This methodology must be adapted to the OLAP analysis pattern. The main contributions of the paper are the following:

- We provide a graph-based model of OLAP analysis; a user analysis consists in a succession of analysis contexts. We model an analysis context through an internal view irrespective of data visualization form.
- Motivated by recommendation techniques in the web field, we define a flexible recommendation paradigm according to details provided by the user.
- We introduce a model of user preferences in OLAP that depend on the analysis context and we discuss a preference-based approach that applies recommendations.

The remainder of the paper is organized as follows: section 2 sets the stage by providing an overview of the decision-support analysis; section 3 introduces promising recommendations in OLAP; section 4 presents a preference-based recommendation approach. Finally section 5 concludes the paper with directions for future research.

2 Decision-Support Analysis

Analytical power of the OLAP technology comes from its underlying multidimensional data model, called constellation [11, 18].

2.1 Multidimensional Data Model

A constellation regroups several facts, which are studied according to several analysis axes (dimensions) possibly shared between facts. It extends star schemas [11], which are commonly used in the multidimensional context.

Definition. A *constellation* is defined as (N^C, F^C, D^C, $Star^C$) where N^C is a constellation name, F^C is a set of facts, D^C is a set of dimensions, $Star^C$: $F^C \rightarrow 2^{D^C}$ associates each fact to its linked dimensions.

Definition. A *dimension,* noted $D_i \in D^C$, is defined as (N^{Di}, A^{Di}, H^{Di}) where N^{Di} is a dimension name, $A^{Di} = \{a^{Di}_1,\ldots, a^{Di}_u\}$ is a set of dimension attributes, $H^{Di} = \{H^{Di}_1,\ldots, H^{Di}_v\}$ is a set of hierarchies.

Within a dimension, attribute values represent several data granularities according to which measures could be analyzed. In a same dimension, attributes may be organised according to one or several hierarchies

Definition. A *hierarchy*, noted $H^{Di}_j \in H^{Di}$, is defined as (N^{Hj}, P^{Hj}, $Weak^{Hj}$) where N^{Hj} is a hierarchy name, P^{Hj}=<id^{Di}, $p^{Hj}_1,\ldots, p^{Hj}_{vj}$, All> is an ordered set of attributes, called *parameters*, which represent useful graduations along the dimension, $\forall k \in [1..v_j]$, $p^{Hj}_k \in A^{Di}$. The $Weak^{Hj}$: $P^{Hj} \rightarrow 2^{ADi-ParamHj}$ function associates each parameter to a set of *weak attributes* for adding semantic information to the parameter.

A fact reflects information that has to be analyzed through one or several indicators, called measures.

Definition. A *fact*, noted $F_i \in F^C$, is defined as (N^{Fi}, M^{Fi}) where N^{Fi} is the fact name, $M^{Fi}=\{f_1(m^{Fi}_1),\ldots, f_w(m^{Fi}_w)\}$ is a set of *measures* associated to aggregation functions $f_1,\ldots,f_p$.

The following figure shows an example of a constellation that allows analysing online sales as well as the purchase activity of a worldwide distributor (graphical notations are inspired by [8]).

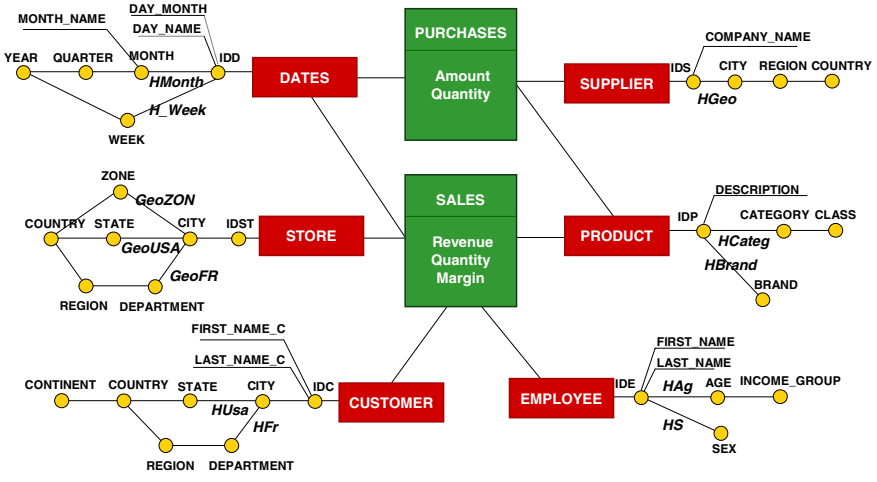


Fig. 1. Example of constellation schema

2.2 OLAP Analysis Modelling

OLAP systems offer capabilities to interactively analyse the data by applying a set of specialized operations, such as drill-down, roll-up and slice-and-dice [1,9,18]. It has been recognized [6,19] that the workload of an OLAP application can be characterized by the user's navigational data analysis task: the user defines a first query then successively manipulates the results applying OLAP operations. Thus, the typical interaction between the user and the system consists of a sequence of queries. Henceforth, the set of queries necessary to answer a business question (*e.g.* "which products are selling abnormally in low quantities?") is referred to as *analysis*. Each query result within a given analysis represents an *analysis context*.

2.2.1 Analysis Context

Within an OLAP analysis both structures and data are displayed. In the context of our research, the term *analysis context* refers to all the items (structures as well as data) that are displayed in a given instant of the analysis.

We model the analysis context through a set of multidimensional structures and values of displayed parameters and measures, called *context components* [10]. We distinguish two categories of context components: 1) components that are related to the fact context C^F: fact (F), measure (m), a value of a measure (val_m) and aggregate function (f^{Agreg}); and 2) components that are related to the dimension context C^D: dimension (D), parameter (p) and a value of a parameter (val_p). Note that each analysis context consists of one fact context and at least two dimension contexts.

Definition. An OLAP analysis context is defined as $\{C^F, C^{D1}, \ldots, C^{Dn}\}$ where

- $\mathbf{C^F} = F\ (/\ f^{Agreg}(m_j) \in \{val_m\})^+$ is a fact context, where f^{Agreg} (AVG, SUM, …) is an aggregate function, $m_j \in M^F$, and $val_m \in Dom(m_j)$,
- $\mathbf{C^{Di}} = Di\ (/\ p_k \in \{val_p\})^+$ is a dimension context, where $p_k \in A^{Di}$ and $val_p \in Dom(p_k)$. Note that attributes of a dimension context must belong to the same hierarchy $H \in H^{Di}$.

Although decisional data are usually displayed within visualization structures that support interpretation and decision making, such as Multidimensional Tables (MT) and charts, the internal view of data is a tree structure.

Analysis Context Tree. An analysis context is expressed by means of a tree *T(V, E)* (where *V* is the set of nodes and *E* is the set of edges) that reflects the nature of the relationship between the components of an OLAP analysis. There are two types of nodes in *V*:

- structure nodes, one for the analysed fact (the root of the tree), one for each analysis indicator (a measure associated with an aggregate function), one for each analysis axis (a displayed dimension), and one for each displayed attribute.
- value nodes, one for each value of an attribute or a measure.

Example 1. Fig. 2 depicts an example of a 2-dimensional analysis context, which displays sales revenue by year according to the countries and cities of customers: C = $\{C^F, C^{D1}, C^{D2}\}$, where C^F = *SALES* / Sum (*Revenue*)∈ {14, 13, 8, 9, 16, 15, 12, 9}, C^{D1} = *CUSTOMER.HFr* / *Country*∈ {France,USA} / *City*∈ {Paris, Toulouse, N-Y, Washington}, and C^{D2} = *DATES.HMonth* / *Year*∈ {2007, 2008}.

The internal view of this analysis context represented in Fig. 2 (a) is displayed to the user according to a MT (*cf.* Fig. 2 (b)) and a chart (*cf.* Fig. 2 (c)).

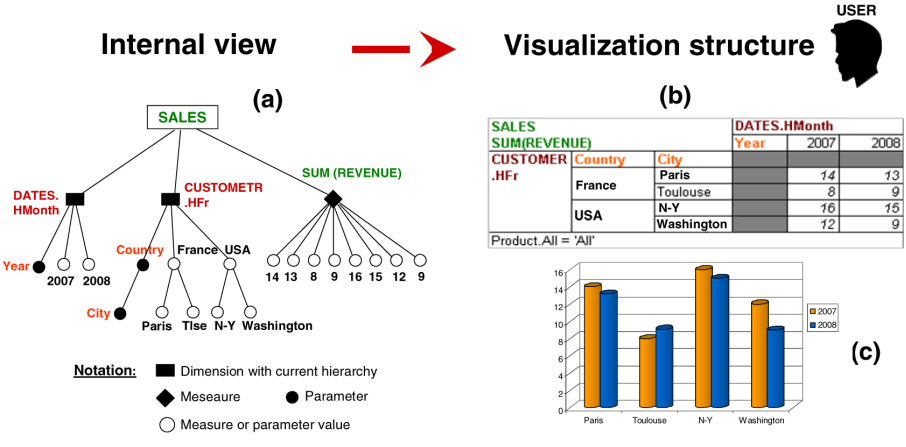


Fig. 2. Example of a context of analysis of sales revenue

2.2.2 OLAP Analysis Graph

An analysis context represents a given state of the OLAP analysis. Therefore, we consider an OLAP analysis as a succession of analysis contexts. User performs OLAP operations to move from one context to another. This navigational pattern is best described by a graph representation, where the current analysis context corresponds to a node. The edges represent transitions between analysis contexts (see Fig. 3).

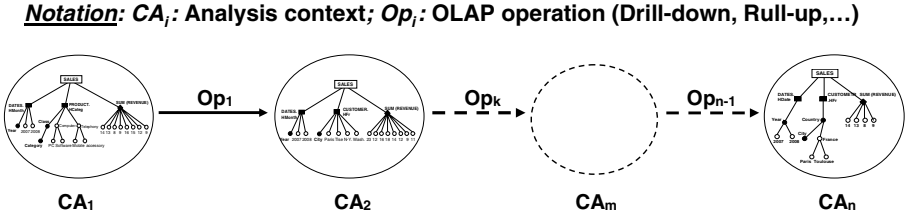


Fig. 3. OLAP Analysis graph

3 Flexible Recommendations in OLAP

In this section, we discuss what should the system recommend to the analyst in order to assist him in navigating through multidimensional data.

A common scenario for existing recommendation systems is a Web application with which a user interacts. The system helps the user to select items during his exploration of online catalog. Actually, the user can specify only some details about products and the system displays items with these details but that are close to his profile. Even if the user sets a full request (with all characteristics of products he is searching for), the system displays products that correspond to his request but it also pops up beside to provide advice on alternative items that seem to interest him.

Existing web recommendation techniques can be categorized according to the level of involvement of the user in the data seeking process (see Table 1). We adapt these existing techniques to OLAP in order to make the navigation process handy and to help users quickly find relevant data.

Table 1. Recommendations in web applications Vs. Recommendations in OLAP

Application features	**E-commerce website**	**OLAP**
Data structure	Transactional Databases	Multidimensional Databases
Data view	Detailed, flat, relational	Summarized, multidimensional
Query process	Isolated queries	Navigational analysis process
User input	**Recommendation output**	
Full query	Additional products close to user profile	- Anticipatory analysis node - Alternative analysis node
Partial query	Products with features stated by user and that are near to his profile	Analysis nodes close to user request and in accordance with his profile

We adopt the graph-based representation of OLAP analyses as the basis for our approach to apply recommendations in OLAP. We define three categories of recommendations according to details provided by the analyst.

(1) Interactive Assistance in Querying Multidimensional Data. End-users analyse multidimensional data using a textual or a graphic language [4, 18]. In the last case, query specification is done implicitly by dragging elements from a navigation zone into the visualization structure (*e.g.*, the MT) and incrementally refining the view. By using either a textual or a graphical language, user must state several details for each query, *i.e.*, the analysis axes, the analysis perspectives, etc. In order to make the MDB querying easier and faster, the user should be guided along the query specification process: the system expands incrementally the query according to user manipulations. For example, the system proposes within a drop-down list appropriate granularity levels when the user specifies the analysis axe, *i.e.*, the system generates recommended items to assist users through their interactions. Otherwise, the system can cope with any request regardless of its conciseness. As a consequence, user can rely upon the system by defining queries which lack details in order to perform his analysis faster. The system allows answering user partial queries; it displays decisional data that are related to user request and that are of particular interest to him. Using such an assistance paradigm will effectively reduce user uncertainty in the discovery of relevant information when navigating the data from a constellation.

(2) Anticipatory Recommendations. This category of recommendations allows reducing user manual effort and the time spent in analysis by anticipating user navigation strategy. Let us consider a user who is interested in detailed data according to days when weekly revenue exceeds 10k Euro. According to the basic "philosophy" of OLAP technology, the user starts by asking for sales according to weeks. Then the user focuses on weeks where sales revenue exceeds 10k Euro. After that, he/she performs a drilling operation along the temporal dimension to see data by days. By keeping a repository of user analysis habits, the system will be able to anticipate the user analysis strategy by displaying data by days as a result to his first query, *i.e.*, the system avoids intermediate states among user analysis and displays directly the relevant analysis node, called *anticipatory* node.

(3) Alternatives Recommendation. The installation of recommender systems in OLAP guides analysts by offering them helpful alternatives that may be interesting for their decision-making process. This type of recommendation provides an alternative node according to the user navigation graph form. The recommended *alternative* nodes are provided in addition to the classic result of a user query. They can be subdivided into three major classes: 1) elaborated analysis nodes, which contain more detailed information comparing to the classic node, 2) missed analysis nodes, the system reminds the user of nodes he should ask for, and 3) other analysis nodes, that represent additional nodes the user did not ask for but that may be interesting for him (*e.g.*, the recommender system may provide useful patterns based on user behavior in similar analysis contexts).
The additional analysis nodes are useful for data interpretation and ease users to better understand the classic result.

In summary, taking into account the interactive and navigational nature of the user query behaviour, applying recommendations in OLAP consists in:

– helping users perform an analysis node; the system provides advice on relevant components (dimension, parameters,…) (Fig. 4, step (1)), and
– suggesting relevant analysis nodes; the system guides users toward relevant patterns by proposing them anticipatory nodes (Fig. 4, step (2)) and even alternative nodes (Fig. 4, step (3)).

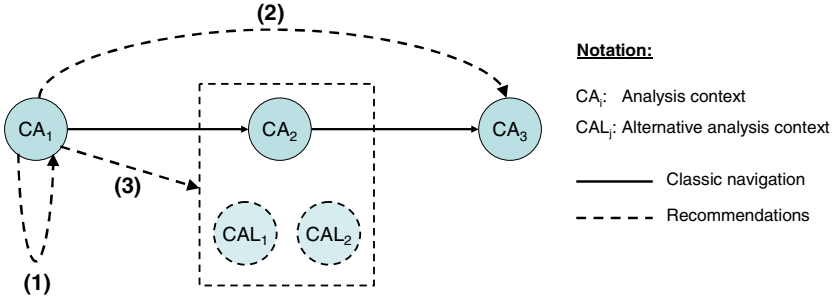


Fig. 4. Applying recommendations upon an OLAP analysis graph

4 Preference-Based Recommendation Framework

Content-based recommenders build on the intuition "find me things like I have liked in the past". Following a content-based approach in OLAP could consider that each analysis node is represented by the set of multidimensional data that it displays (analysis context), and each user is represented by a list of analysis preferences.

In the following subsections we describe our user preference model and show how such a model can be used to generate recommendations.

4.1 User Preferences Modelling

Analysts have various preferences determined upon different analysis contexts [10]. We consider two main categories of user preferences: preferences relating to the analysis axes and preferences concerning the analysis precision.

4.1.1 Preference Context

The user may have preferences that depend on more or less general contexts, *e.g.* a user preference can be associated with the context of analysis of sales or with a more detailed context such as the analysis of sales of a given product category. A preference context C^P is a fragment of the analysis context tree. Actually, the context of a user preference does not necessarily contain all the analysis context components. This can be expressed by assigning the value *all* to the corresponding context components. For example, a user preference that is associated with the context of analysis of sales can be applied in every analysis of sales data irrespective of the analysis axes and parameters. The more the preference context is detailed, the more the related user interest is specific.

4.1.2 Contextual Preference Model

A preference between dimensions defines relevant dimensions for the fact analysis in a specific context.

Definition. Given a constellation C, a preference between dimensions, noted $P^C_k = (\succ_p, C^P)$, is a strict partial order over the subset of the constellation dimensions that are connected to the same fact, where $\succ_p \subseteq D^C \times D^C$ and C^P is a preference context.

A preference within a dimension provides priority parameters (dimension attributes) for data analysis in a given context.

Definition. A preference within dimension D, noted P^H_k, is defined as $(\succ_p, C^P)$ where

- $\succ_p$ is a strict partial order over $A^H \subset A^D$, where A^H is a set of parameters and weak attributes situated on the hierarchy H of dimension D and $\succ_p \subseteq A^H \times A^H$
- C^P is a preference context

Example 2. The decision-maker prefers to analyze sales revenue in priority by country then by region, but he/she may also wish to see more detailed data according to cities then by country in the context of analysis of yearly sales revenue. Such analysis preferences within dimension *Customer* are defined as follows:

- P^{HFr}_1: *Country* $\succ_p$ *Region*, C^{P1} = {*Sales*/ Sum(*Revenue*)}
- P^{HFr}_2: *City* $\succ_p$ *Country*, C^{P2} = {*Sales*/ Sum(*Revenue*), *DATES.HMonth* / *Year* }

This model deals with several user preferences within a dimension (respectively between dimensions) that are related to parameters of the same hierarchy (rep. related to the same fact) providing that they depend on different contexts. Otherwise, for a given analysis context, in the case of a single preference on parameters of a dimension D, their hierarchy will be considered as a hierarchy by default for D exploration. If there are more, a conflict between hierarchies arises: what hierarchy should be considered to explore D? Hence, it is necessary to define a priority order between hierarchies (preference between hierarchies) to solve such kind of conflict.

Definition. Given a dimension D, a preference between hierarchies is a strict partial order $P^D_k = (D, \succ_p)$, where $\succ_p \subseteq H^D \times H^D$.

We call the set of contextual preferences that hold for a MDB, *profile P*. By *CP*(*P*), we denote the set of preference contexts C^P that are associated with at least one preference in *P*. We assume that such profiles are available. In practice, users may express their preferences explicitly. These preferences may be also mined from the previous behavior of the users.

4.2 Recommendation Generation

Contextual preferences are used to retrieve relevant analysis elements which are then used to generate recommendations for the user.

4.2.1 User Preference Selection

User preferences are used to enhance the current analysis context or to build additional contexts that are near to the displayed context.

Now given the current analysis context *CA*, we would like

(1) to identify the set $PCand \subseteq P$ of preferences (P, C^P) for which $C^P = CA$, and then,
(2) use them to enhance *CA or* to generate recommended analysis contexts according to *CA*.

For a given context *CA*, there may be no preference (P, C^P) in the profile *P*, with $C^P = CA$, that is $CA \notin CP(P)$. Actually, the profile contains preferences that do not necessarily depend on all analysis context components. To address this, we use those preferences in *P* that depend on *CA*, *i.e.*, preferences whose contexts are included in *CA*.

The problem of preference selection is a problem of trees matching [10]: a preference whose context tree is included in (all its edges and nodes belong to) the tree of the current analysis context is a candidate preference. If there are several candidate preferences, the selected preference is the most relevant one; it's the preference whose context covers more the current analysis context; *i.e.* whose context tree has the largest number of nodes.

Depending on the type of the preference context C^P, we distinguish two cases:

– C^P concerns a value of a measure or a parameter: integrating the underlying preference leads to move on to a next analysis node. For example, a user prefers to see detailed data according to days when analysing the sales revenue in Italy (C^P = {*Sales*/ Sum(*Revenue*), *Customer.HFr* / *Country* = 'Italy'}). When focusing on revenue in Italy, the user needs to turn on the analysis of data by months.
– C^P does not contain values: integrating the related preference allows enhancing the analysis context.

4.2.2 Computing Recommendations

A recommender system maintains a repository of user preferences that are used to suggest relevant patterns. Hence a key question becomes how does recommender system use these preferences to compute recommendations?

An OLAP recommender system allows users to perform full or partial queries and to ask for help to build their analysis report.

User Partial Query. A partial query generates an incomplete analysis context which can not be displayed to the user. For each query, the system builds a recommendation in an ascending way by enhancing the analysis context resulting from the user query until it becomes well-rounded: (i) filling-out favourite dimensions for the current fact analysis, (ii) specifying the relevant granularity levels of each dimension (dimension parameters), and (iii) aggregating fact data according to the specified parameters.

Example 3. The marketing manager analyses yearly sales revenue according to products' categories and classes (see CA_1 in Fig. 5). He/she intends to change the *Product* dimension by *Customer* axis. Although he/she does not specify the granularity level within the *Customer* axis, the system generates a complete analysis context which is close to his preferences. The system takes into account the user preference $P^{HFr}{}_2$ (see example 2) to enhance the intermediate context (see CA_{interm} in Fig. 5). Actually, both $P^{HFr}{}_1$ and $P^{HFr}{}_2$ are candidate preferences but C^{P_2} covers more the current context CA_{interm}.

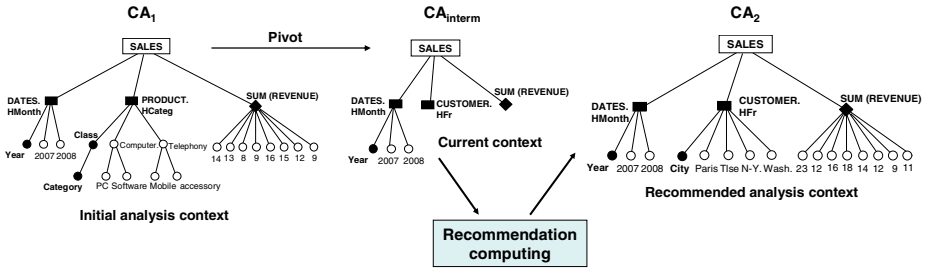


Fig. 5. Partial analysis context expansion

User Full Query. When the user performs a full query, the system computes the query result which will be considered as the current analysis context of user. Then it looks for extra data patterns (*i.e,* extra analysis nodes) that are interesting to the user in this current analysis context. Following a preference-based paradigm, such analysis nodes are dynamically made up according to user preferences.

The basic idea is to gradually construct analysis contexts by altering the current analysis context through preferences integration. Preferences that are related to the current context are integrated in decreasing order of their degrees of hierarchy:

- The system searches for preferences between dimensions to change a current dimension by other relevant one. Then, parameters are specified through preferences within the selected dimension.
- The system changes current parameters according to preferences within current dimensions. The generated analysis context differs from the classic one by the granularity levels.

4.3 Recommendation Display

The system determines recommended analysis contexts (internal view), then displays them to the user according to the visualisation structure he uses. Recommendations are provided to the user according to their types:

- An anticipatory recommendation is displayed instead of a classic result. The user is enabled to customize the system by stating whether he wants to authorize such recommendation. An explanation for anticipatory recommendation is displayed besides in order to establish trust in the recommender system.
- Recommended alternativess are displayed in a separated part of the visualization interface. Only alternative dataset prototypes (data structure, *i.e.*, fact, measures, dimensions, parameters and restriction predicates) are displayed to the user. The system loads dimension data (parameters values) as well as fact data (measures values) when the user selects a recommended prototype.

4.4 Example

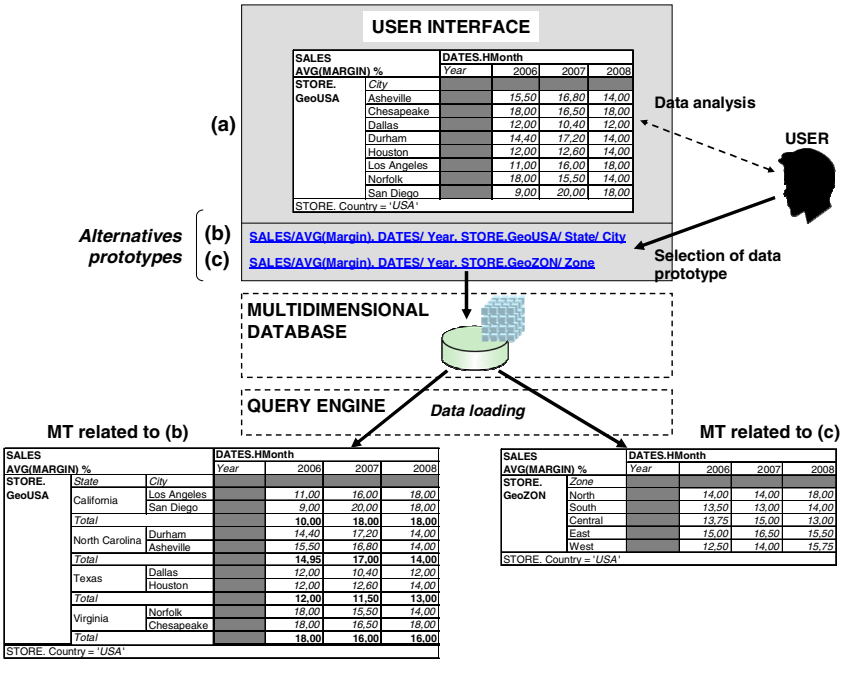
Let us consider a decision-maker who has the following preferences that are deduced from his previous interactions with the system:

– P^{GeoUSA}_{3}: State $\succ_p$ City, C^{P3} = {*Sales*/AVG(*Margin*), *Dates.HMonth* / *Year*}
– P^{GeoZON}_{4}: *Zone* $\succ_p$ *Country*, C^{P4} = {*Sales*/AVG(*Margin*)}
– P^{Store}_{5} : *GeoZON* $\succ_p$ *GeoUSA*

Suppose that the user intends to analyse profit margin in USA according to dimensions *Store* and *Dates*, especially by *City* (from hierarchy *GeoUSA*) and by *Year*. The system displays the classic query result (see Fig. 6 (a)). Furthermore, it provides two alternative recommendations in order to help the user quickly find relevant information and discover interesting patterns.

The first alternative (profit margin by year according to cities and states, see Fig. 6 (b)) is generated since the user is interested in data by state in the context of analysis of yearly profit margin (P^{GeoUSA}_{3}). This analysis node is richer on information than the classic node since it provides more details on cities (state of each city). It provides also correlations between cities themselves, *i.e.,* the user can check the effectiveness of the values related to each city by observing its margin part in the total margin of its state. This may help user evaluate data.

The second alternative (profit margin by years and by Zone, see Fig. 6 (c)) provides another perspective to analyse the profit margin. The user is interested in data according to the geographical perspective (*GeoZON* according to P^{Store}_{5}) and more precisely according to the level *Zone* in the context of analysis of profit margin (P^{GeoZON}_{4}).



SALES AVG(MARGIN) %			DATES.HMonth Year	2006	2007	2008
STORE. GeoUSA	City					
	Asheville			15,50	16,80	14,00
	Chesapeake			18,00	16,50	18,00
	Dallas			12,00	10,40	12,00
	Durham			14,40	17,20	14,00
	Houston			12,00	12,60	14,00
	Los Angeles			11,00	16,00	18,00
	Norfolk			18,00	15,50	14,00
	San Diego			9,00	20,00	18,00

STORE. Country = 'USA'

SALES AVG(MARGIN) %			DATES.HMonth Year	2006	2007	2008
STORE. GeoUSA	State	City				
	California	Los Angeles		11,00	16,00	18,00
		San Diego		9,00	20,00	18,00
	Total			**10,00**	**18,00**	**18,00**
	North Carolina	Durham		14,40	17,20	14,00
		Asheville		15,50	16,80	14,00
	Total			**14,95**	**17,00**	**14,00**
	Texas	Dallas		12,00	10,40	12,00
		Houston		12,00	12,60	14,00
	Total			**12,00**	**11,50**	**13,00**
	Virginia	Norfolk		18,00	15,50	14,00
		Chesapeake		18,00	16,50	18,00
	Total			**18,00**	**16,00**	**16,00**

STORE. Country = 'USA'

SALES AVG(MARGIN) %		DATES.HMonth Year	2006	2007	2008
STORE. GeoZON	Zone				
	North		14,00	14,00	18,00
	South		13,50	13,00	14,00
	Central		13,75	15,00	13,00
	East		15,00	16,50	15,50
	West		12,50	14,00	15,75

STORE. Country = 'USA'

Fig. 6. Framework for alternatives recommendation

5 Conclusions

We proposed to apply recommendations in OLAP systems in order to assist the user during his decision-support analysis. This includes both implicit assistance in the form of anticipatory recommendations and explicit assistance by providing alternative patterns of data or helping user perform his analysis reports. Our approach deals with recommendation of an analysis context which represents a state of an OLAP analysis. It is independent from user structure visualization. The system determines recommended analysis contexts, then displays them to the users according to their data visualization structure (MT, chart, diagrams,…).

We defined three categories of recommendations in OLAP according to details provided by user and we discussed how recommendations are generated to the user with regard to his preferences.

As a future work, we intend to specify preference mining techniques for detecting strict partial order preferences in user log data. These techniques must: 1) elicit user preferences; and 2) discover mappings that associate the user preferences to their related analysis contexts. We intend also to investigate how to make progressive improvement of the recommendations while a user is increasingly using the system. This leads to conversational recommenders that participate in an interactive dialog with the user by asking him to give feedback or to answer questions.

References

1. Abelló, A., Samos, J., Saltor, F.: Implementing operations to navigate semantic star schemas. In: International Workshop on Data Warehousing and OLAP, pp. 56–62. ACM, New York (2003)
2. Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval. Addison-Wesley, Reading (1999)
3. Balabanovic, M., Shoham, Y.: Fab: Content-based, collaborative recommendation. Communications of the ACM 40(3), 66–72 (1997)
4. Cabibbo, L., Torlone, R.: From a procedural to a visual query language for OLAP. In: International Conference on Scientific and Statistical Database Management, pp. 74–83. IEEE Computer Society, Washington (1998)
5. Choong, Y.W., Laurent, D., Marcel, P.: Computing Appropriate Representations for Multidimensional Data. Data & knowledge Engineering Journal 45(2), 181–203 (2003)
6. Dittrich, J.P., Kossmann, D., Kreutz, A.: Bridging the gap between OLAP and SQL. In: International Conference on Very Large Data Bases, pp. 1031–1042 (2005)
7. Giacometti, A., Marcel, P., Negre, E.: A Framework for Recommending OLAP Queries. In: International Workshop on Data Warehousing and OLAP, pp. 73–80. ACM, New York (2008)
8. Golfarelli, M., Maio, D., Rizzi, S.: Conceptual Design of Data Warehouses from E/R Schemes. In: Annual Hawaii International Conference on System Sciences (1998)
9. Gyssen, M., Lakshmanan, L.: A foundation for multi-dimensional databases. In: International Conference on Very Large Data Bases, pp. 106–115 (1997)
10. Jerbi, H., Ravat, F., Teste, O., Zurfluh, G.: Management of context-aware preferences in Multidimensional Databases. In: International Conference on Digital Information Management, pp. 669–675 (2008)

11. Kimball, R.: The Data Warehouse Toolkit, 1996, 2nd edn. John Wiley and Sons, Chichester (2003)
12. Konstan, J.A., Miller, B.N., Maltz, D., Herlocker, J.L., Gordon, L.R., Riedl, J.: GroupLens: Applying Collaborative Filtering to Usenet News. Communications of the ACM 40(3), 77–87 (1997)
13. Koutrika, G., Ikeda, R., Bercovitz, B., Garcia-Molina, H.: Flexible Recommendations over Rich Data. In: ACM Conference On Recommender Systems, pp. 203–210. ACM, New York (2008)
14. Lieberman, H.: Autonomous Interface Agents. In: SIGCHI Conference on Human Factors in Computing Systems, pp. 67–74. ACM, New York (1997)
15. Linden, G., Smith, B., York, J.: Amazon.com recommendations: Item-to-item collaborative Filtering. IEEE Internet Computing 7(1), 76–80 (2003)
16. Maes, P.: Agents That Reduce Work and Information Overload. Communications of the ACM 37(7), 31–40 (1994)
17. Miller, B.N., Albert, I., Lam, S.K., Konstan, J.A., Riedl, J.: Movielens unplugged: Experiences with an occasionally connected recommender system. In: ACM International Conference on Intelligent User Interfaces, pp. 263–266 (2003)
18. Ravat, F., Teste, O., Tournier, R., Zurfluh, G.: Algebraic and graphic languages for OLAP manipulations. International Journal of Data Warehousing and Mining 4(1), 17–46 (2008)
19. Sapia, C.: PROMISE: Predicting Query Behavior to Enable Predictive Caching Strategies for OLAP Systems. In: Kambayashi, Y., Mohania, M., Tjoa, A.M. (eds.) DaWaK 2000. LNCS, vol. 1874, pp. 224–233. Springer, Heidelberg (2000)
20. Satzger, B., Endres, M., Kießling, W.: A Preference-Based Recommender System. In: Bauknecht, K., Pröll, B., Werthner, H. (eds.) EC-Web 2006. LNCS, vol. 4082, pp. 31–40. Springer, Heidelberg (2006)

Classification and Prediction of Software Cost through Fuzzy Decision Trees

Efi Papatheocharous and Andreas S. Andreou

University of Cyprus, Department of Computer Science
75 Kallipoleos str., CY1678 Nicosia, Cyprus
{efi.papatheocharous,aandreou}@cs.ucy.ac.cy

Abstract. This work addresses the issue of software effort prediction via fuzzy decision trees generated using historical project data samples. Moreover, the effect that various numerical and nominal project characteristics used as predictors have on software development effort is investigated utilizing the classification rules extracted. The approach attempts to classify successfully past project data into homogeneous clusters to provide accurate and reliable cost estimates within each cluster. CHAID and CART algorithms are applied on approximately 1000 project cost data records which were analyzed, pre-processed and used for generating fuzzy decision tree instances, followed by an evaluation method assessing prediction accuracy achieved by the classification rules produced. Even though the experimentation follows a heuristic approach, the trees built were found to fit the data properly, while the predicted effort values approximate well the actual effort.

Keywords: Software cost estimation, fuzzy decision trees, CHAID, CART, classification.

1 Introduction

Software cost estimation is essentially a managerial process used to assess the total costs spent, including money, physical and technical resources, as well as the time and effort, during the development process of a software product. Regularly, the process of software cost estimation is performed right before the establishment of a new contract agreement. Nonetheless, after the project initiation, the estimate is further refined iteratively in subsequent project phases throughout the whole project's life-cycle. Moreover, software development companies, as well as their stakeholders, such as end-users, customers, managers and researchers, support that accurate and consistent estimation of the required effort spent for developing software is a prerequisite in software engineering for successful project completion [1]. Such assessments are considered more useful for better allocation of project resources if attained at the initial project phases. At the early development phases, software cost estimation is difficult to perform as the situation is fluid and there is high uncertainty over the project parameters. Moreover, the project characteristics are more likely to change and contingencies affect the development process, as there are many

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 234–247, 2009.
© Springer-Verlag Berlin Heidelberg 2009

constraints still undefined. According to Boehm and his 'cone of uncertainty' [2] the amount of uncertainty and the associated risk to the cost estimates decreases along with the project's progression to completion [3].

A major focus of researchers and an increasing interest by practitioners since the 1980s to reduce the aforementioned uncertainty has led to the development of many software cost estimation models. Numerous research studies increasingly recognize the importance of successful project planning and estimation, however, despite these efforts there is lack of confidence in the results reported. Empirical studies investigate the efficacy of the methods used and the impact of various factors on productivity, quality and cost, but, at the same time, different studies frequently report controversial findings [4]. The lack of terminology and consistency in describing experiments and results as noted in several surveys (e.g., [5], [6]) result in disparate conclusions reported within and among different techniques, approaches, measures, factors, datasets and researchers investigating the potential of developing accurate software cost prediction systems. Moreover, the situation of the average effort overrun has not been considerably improved over the last 10-20 years [7]. It seems that the dynamic nature of the software process often hinders approaches to accurately and robustly estimate effort, while the process remains highly prone to human errors and biases [8].

Information available at the early phases of development is mostly of a categorical nature (i.e., linguistic values) that is not however considered an adequate source for safe estimations. Such information includes the type of the organization, the culture and stability of the development environment, the business area, the application domain, the development platform and the language type used for the project. Even though this information is usually known prior to the project initiation, and project stakeholders may somehow predict the average number of people required to work on the project, as well as the project size, the problem is how to include such 'linguistic' information in the cost model.

This paper aspires to address the need for approximating software cost estimation as accurately as possible and, at the same time, understand and possibly tackle the inherent uncertainty of the estimation process especially when dealing with information of categorical nature. In this context, what we propose is the utilization of common machine learning structures, such as Decision Trees, enhanced by Fuzzy Logic, to provide a bounded estimation of cost within a range of values. The novelty of our approach lies with the automatic generation of instances of Fuzzy Decision Trees (FDT) that yield robust classification rules with high confidence levels. The classification rules are further exploited to gain statistically accurate and reliable cost estimates for new project data.

The rest of the paper is organized as follows: Section 2 briefly presents an overview of software cost estimation literature and discusses the recent emergence of hybrid methods combining machine learning and fuzzy logic techniques. Section 3 describes the proposed method for creating and applying Fuzzy Decision Trees (FDT) on a group of selected and pre-processed cost data schemes. A detailed description of the experiments is provided in Section 4 followed by the prediction results obtained. Finally, Section 5 presents our conclusions, along with some suggestions for improving the methodology, and discusses some future research steps.

2 Brief Literature Review

Over the last twenty years software cost estimation research has demonstrated extensive interest in the application of a number of techniques used to create various effort prediction models. Among them, classical approaches involve algorithmic models and estimations by analogy. The former attempts to introduce mathematical equations containing various cost factors for approximating the associated development effort usually via regression techniques and may also use expert judgment (examples of which are COCOMO I [2], COCOMO II [9], Function Points Analysis [10], SLIM [11] etc.). Estimation by analogy takes advantage of similarity measures and attempts to relate knowledge or historical samples with the characteristics of the new project so as to guide estimation towards the value of the closest match. More recently, machine learning approaches have been proposed to enhance the above. Examples include artificial neural networks [12], case-based reasoning, decision tree learners ([13], [14], [15]), genetic programming [16] etc.

Lately, various hybrid forms of techniques and cost models attempting to improve intuitiveness, accuracy and robustness emerged in an increasing number of studies [6]. Especially, techniques using concepts of fuzzy logic have gained much interest among researchers [1]. In addition, researchers suggest that data-driven techniques applied in combination with a multiple set of techniques on different subsets of data may produce a range of estimated values instead of crisp values and may reduce the inaccuracy degree involved in the estimation ([17], [18]). Consequently, the notion of prediction interval was introduced as a minimum-maximum range of values for the effort estimates, attached with a confidence level that the actual value of the effort will be included in the range [7]. Previous studies reported improvement in classical data mining structures like decision trees when fuzzification was applied ([19], [20]).

Taking previous suggestions into consideration the creation of Fuzzy Decision Trees (FDT) may be conceived as a promising solution in classifying project data and extracting association rules describing the nature of the software development environment. Our approach aspires to promote adaptive and dynamic automated mechanisms to reach to better solutions in such approximation problems. In order to alleviate the deficiencies of the techniques proposed in the relative literature, we decided to address the problem in a possibly more effective and more interpretable manner with the FDT methodology proposed which conceptually interprets the encrypted information from a large heterogeneous dataset in such a way that can be comprehensively understood by individuals.

3 Methodology

From our point of view a data-driven method to develop software cost models typically describes and analyzes the project characteristics of completed projects stored in a database and focuses on explaining the factors that influence effort. Additionally, given a degree of symmetry, in either analogy or resemblance, between the current project for which we wish to estimate its cost and a reduced set of past projects, we may apply a transformation on the past effort values to statistically approximate the actual effort required for the new project. In our case, the methodology we propose is graphically depicted in Figure 1 and consists of three

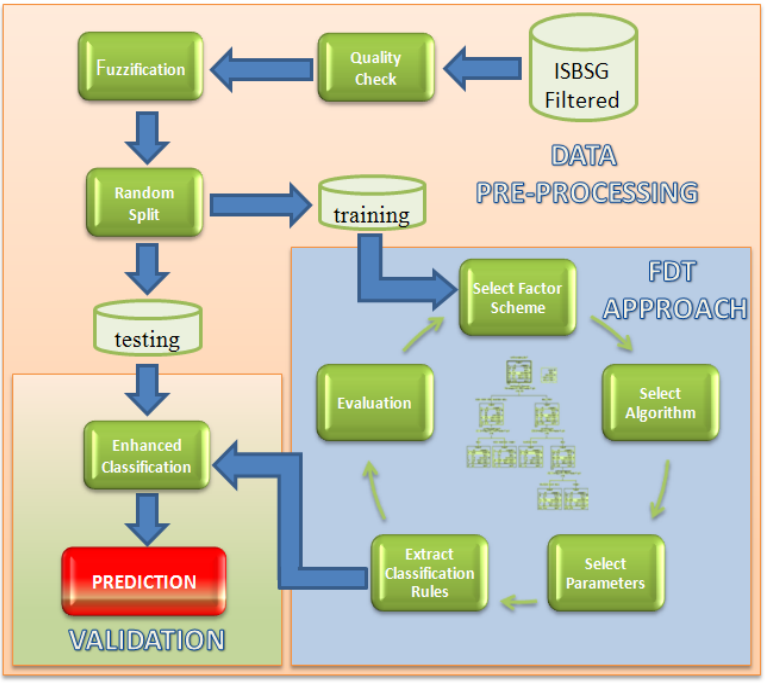


Fig. 1. The proposed FDT methodology

stages: (i) Data Pre-processing and Fuzzification, (ii) Training with Fuzzy Decision Trees (FDT), including creation and evaluation, and (iii) Prediction with the Classification Rules obtained and Class Resemblance Prediction enhancement, along with Validation activities.

It is clear, therefore, that the proposed methodology attempts to combine and compare, to some extent, various techniques for building an improved software cost estimation model. The techniques involve a combination of Fuzzy Logic theory [21], Decision Trees [22], and more particularly the Chi-squared Automatic Interaction Detection (CHAID) [23] and Classification and Regression Trees (CART) [24] algorithms. The detailed description of the three stages of our methodology is given in the next sections.

3.1 Cost Factors Description

The ISBSG dataset [25] used in the experiments consists of various software project cost records coming from a broad cross section of countries and range in size, effort, platform, language and development technique data. This dataset may be characterized as highly heterogeneous and may even contain biased samples due to the diversity of the methods used to measure mostly size and final effort count, along with the interpretation of the measurement process as described by the Group. The dataset initially contained 92 variables from which a reduced number of project attributes was selected. More specifically, factors that expressed information derived from other

factors, or factors that were considered insignificant to effort were excluded from the final dataset used (from now on the new reduced dataset will be referred to as 'ISBSG Filtered'). The factors selected reflect process, product and people characteristics and are summarized in Table 1. The Full Cycle Work Effort (EFF) is the dependent variable; the next five factors on the table, sequentially from top to bottom, are of numerical type and the last five of categorical type. Additionally, factors participating in Table 1 excluding EFF, PET, PIT and PDRU may be considered as available and thus be measured from the early phases of the project life-cycle.

Table 1. Summary of selected software cost factors

ID	Factor	Description
EFF	Full Cycle Work Effort	Total effort (in hours) recorded against the project
PET	Project Elapsed Time	Total elapsed time for the project (in calendar months)
PIT	Project Inactive Time	The number of calendar months in which no activity occurred
PDRU	Project PDR (ufp)	Project delivery rate (in hours per function point) equals to the quotient of effort and functional size
AFP	Adjusted Function Points	Functional size of the project at the final count
ATS	Average Team Size	Average number of people that worked on the project
DT	Development Type	Description of whether the project is a New Development, Enhancement or Re-development
AT	Application Type	Description of the application addressed by the project
DP	Development Platform	Description of the primary development platform (e.g., PC, Multi-Platform etc.)
LT	Language Type	Definition of the language type used by the project (e.g., 3GL, 4GL, etc.)
RL	Resource Level	Describes the four levels about the people whose time is included in the work effort data reported

3.2 Fuzzy Decision Trees and Classification Rules

The FDT structure selected for modelling software effort employed the CHAID (Chi-squared Automatic Interaction Detection) and CART (Classification and Regression Tree) algorithms. Such non-parametric structures have the ability to model complex problems, handle noisy (or outlying) data and missing values, and work with categorical data. Additionally, the major advantage over other machine learning approaches is that FDT may produce predictive modelling tools offering a degree of confidence and yielding self-descriptive rules, thus considered easier to interpret by individuals.

Classic decision trees involve nodes in a flow-chart, structure with the top-most node called 'root' and the terminal nodes called 'leaves'. The internal nodes represent attributes and each branch corresponds to an outcome of the test performed on a specific attribute. The basic algorithm for decision trees follows a top-down recursive divide-and-conquer way to build the tree. The process selects the best criterion to split

the data (from the root of the tree to the child nodes) and iteratively continues until the tree is created as homogeneously as possible and until a stopping criterion is satisfied. The present work proposes FDT in which fuzzification is performed on the participating variables prior to building the tree, while the dependent variable (EFF) is placed at the root. Then, applying various criteria the data is split into groups and utilizing the greedy local search method optimal FDT structures are pursued. Specific branches are pruned and finally the algorithm terminates yielding tree structures. An example of such a structure is illustrated in Figure 2. This tree is interpreted by rules of the form "*If (condition 1 AND condition 2 AND … AND condition N) then Z*", where the conditions are extracted from the nodes and *Z* is the root. Each path from the root node to a terminal node corresponds to a fuzzy rule.

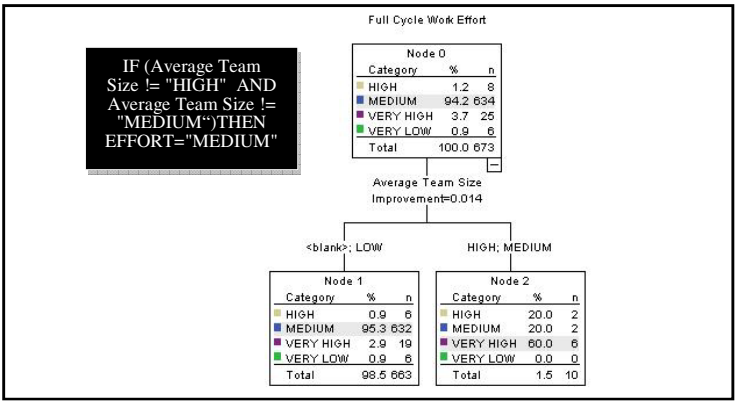


Fig. 2. An example of a FDT and an indicative rule

3.3 Design of the Experiments

This section describes the detailed design and execution of a set of experiments that demonstrate the steps of the methodology and assess its validity. Initially, the data went through some pre-processing and fuzzification activities and then the proposed FDT creation process was followed. The experiments conducted were iteratively executed changing several internal parameters of the algorithms in each iteration to create robust FDTs and produce a set of unique strong rules. The available data records were split into two subsets; the first one was used to construct the FDT, thus called the training set. The second subset, called testing set, was used to test the produced FDT and to assess the efficiency and generalization of the corresponding rules on the tree. Within the same phase two threshold rules were applied to examine whether the technique may be used to maximize the homogeneity within clusters of projects and improve effort prediction.

3.3.1 Data Pre-processing and Fuzzification

To begin with, the 'ISBSG Filtered' dataset went through various integrity checks and filtering to preserve quality by deleting the projects rated by ISBSG as 'C' or 'D' which correspond to low integrity data. Also, we adopted the suggestion of the

ISBSG's quality reviewers assessing the integrity of the application of the Functional Size Measurement Method (e.g., IFPUG, MARK II, NESMA, COSMIC-FPP etc.) and kept only the projects measured with 'IFPUG'. Additionally, the Summary Work Effort of the projects measured only for the full development life-cycle was taken into consideration and not those measured over a few of the phases.

All numerical cost factors described in Table 1 were fuzzified and in addition, the dependent variable (EFF) was normalized by the natural logarithm to decrease the model's output value ranges that will be later used for prediction. Furthermore, the probability distribution of each cost driver indicated that the ordinal transformation of each variable can be generated and exploited using the trapezoidal membership function. The numerical values of the attributes whose value ranges were relatively large (i.e. of the order of thousands) were separated into 5 ordinal intervals of equal size, whereas those that had small value ranges (i.e. of the order of hundreds) were separated into 3 ordinal intervals of equal size. This conversion to ordinal intervals was clearly empirical following common practices adopted in similar cases in classic fuzzy logic literature (see Table 2). Therefore, the EFF and AFP attributes are measured on an ordinal scale of five linguistic values (from 'very low' to 'very high') and the rest four attributes on a scale of three linguistic values (from 'low' to 'high').

Table 2. Fuzzy interval values

ID	FUZZY VALUES				
	VERY LOW	**LOW**	**MEDIUM**	**HIGH**	**VERY HIGH**
EFF	≤ 3.90	3.91-5.73	5.74-7.55	7.56-9.37	≥ 9.38
AFP	≤ 3506	3507-7010	7011-10514	10515-14017	≥ 14018
PET	-	≤ 17.23	17.24-34.65	≥ 34.66	-
PDRU	-	≤ 129	129-258	≥ 258.10	-
PIT	-	≤ 4.0	4.10-7.90	≥ 8.0	-
ATS	-	≤ 26.29	26.30-51.62	≥ 51.63	-
EFF	≤ 3.90	3.91-5.73	5.74-7.55	7.56-9.37	≥ 9.38
AFP	≤ 3506	3507-7010	7011-10514	10515-14017	≥ 14018

The input attributes were fuzzified by determining the degree to which they belong to each of the appropriate fuzzy sets via membership functions. For each cost attribute variables m_i, n_i, a_i and b_i were calculated ($1 \leq i \leq n$, and n is the number of linguistic terms in the classification table being analyzed) according to equations (1)-(4) and following the fuzzification illustrated in Figure 3 [26].

$$m_i = \min_{\substack{value\ of\ linguistic\ term\ T_i \\ in\ classification\ table}} \tag{1}$$

$$n_i = \frac{m_i + m_{i+1}}{2} \tag{2}$$

$$a_i = n_{i-1} \tag{3}$$

$$b_i = m_{i+1} \tag{4}$$

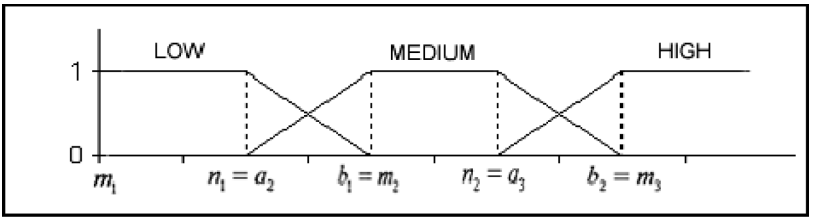


Fig. 3. Membership function of the fuzzification

Finally, the dataset was separated into training and testing as previously mentioned, with 70% and 30% random sampling of the total dataset respectively (no common data samples). The former was utilized during construction of the FDT and the latter for evaluating their cost estimation ability by validating the rules extracted and providing an initial mean effort value prediction based on the intervals of the projects satisfying the rules.

3.3.2 FDT Creation and Rule Extraction

The experimental approach applies iteratively a heuristic technique to create and evaluate the FDTs. Three different cost Driver Schemes (DS) were created, each consisting of different attributes named *All*, *Categorical* and *Early* and abbreviated as S1, S2 and S3 respectively. The logic behind creating these three schemes of data inputs was to assess the power of descriptive cost factors on effort as this is reflected on the available empirical data samples. Therefore, we decided to include in the first scheme (S1) all cost attributes listed in Table 1, while the second scheme (S2) includes only five categorical attributes (namely DT, AT, DP, LT and RL). Finally, the third scheme (S3) contains only those attributes that are available and can be measured from the early stages of project development (i.e., all attributes except PET, PIT and PDRU). This means that the third scheme may be considered more important than the rest in terms of practical value and usefulness.

In order to select the optimal execution parameters according to the problem examined and the algorithm used (CHAID or CART) we experimented in a repetitive manner evaluating the rules according to the significance level threshold and degree of occurrence obtained. Ultimately, our aim was to reach to stable rules that include a large number of project characteristics and yield specific clusters of projects improving the overall prediction accuracy. To this end, the deepest trees were finally selected for further experimentation and evaluation. Therefore, the representative set of rules displayed in Table 3 was extracted for further processing based on the hypothesis that the most reliable rules are the ones that performed well during training (high significance level) and maximized the homogeneity of the set of projects satisfying the rules, i.e., classifying appropriately the project samples with similar characteristics.

3.3.3 Evaluation

The evaluation of the FDT created at the training phase was based on a combination of measures. Due to the fact that the technique is data driven, several FDT generated were identical and thus the rules that appeared more frequently were considered to better describe the distribution of the samples and thus were reported in the results

section. Each leaf of the FDT indicates a class or effort value range according to the distribution (Significance Level (*SL*)) and is represented by a classification rule as mentioned earlier. The variable that classifies the majority of the training samples is placed at the top of the tree and exhibits the most significant relationship with the dependent variable. Also, besides the statistical significance, the 'goodness' of each rule is evaluated on the number of factors participating in the rule (*NF*), thus reaching to a more homogeneous cluster of data as previously mentioned. The promoted rules are then used for classification and validation.

Most specifically we worked as follows: The numbers of the train and test project samples that satisfy a rule *r* are defined as:

$$n_{train,r} = |N_r| \tag{5}$$

$$n_{test,r} = |L_r| \tag{6}$$

where N_r = *{train project samples that satisfy rule r}* and L_r = *{test project samples that satisfy rule r}*. For each rule *r* satisfied we calculate the mean effort range $\widehat{eff_r}$ using equation (7), with σ_{n_r}being the standard deviation of the respective project samples satisfying rule *r*. Essentially, equation (7) takes the mean effort value of the projects that were classified in a certain cluster according to rule *r* as the predicted effort value of the new project (provided that the new project is also classified in the same cluster), with a deviation tolerance threshold equal to the standard deviation of the projects in the cluster.

$$\widehat{eff_r} = \frac{1}{n_r}\left(\sum_{i=1}^{n_r} eff_{i,r}\right) \pm \sigma_{n_r} \tag{7}$$

For the test samples that satisfy rule *r* we define an additional threshold measure, namely Prediction Measure (*PM*) and define the number of samples that satisfy it as:

$$n_{PM,r} = |C_r| \tag{8}$$

where C_r = *{test project samples in L_r that satisfy inequality (9)}*.

$$\frac{1}{n_{train,r}}\left(\sum_{i=1}^{n_r} eff_{i,r}\right) - \sigma_{n_{train,r}} \leq \widehat{eff_r} \leq \frac{1}{n_{train,r}}\left(\sum_{i=1}^{n_r} eff_{i,r}\right) + \sigma_{n_{train,r}} \tag{9}$$

The Hit Ratio (*HR*) of *PM* is defined as:

$$HR_{PM,r} = \frac{|C_r|}{n_{testing,r}} \tag{10}$$

So far we were able to produce a predicted effort value according to equation (7). We will now attempt to improve this estimation by considering additional factors than those participating in a rule thus improving homogeneity of the associated cluster. Therefore, for each sample *p* in the test set that satisfies rule *r* we work as follows: We define a Resemblance Threshold (*RT*) for each data scheme S1, S2 or S3 as:

$$RT_j = \left\lfloor \frac{ds_j}{2} \right\rfloor \tag{11}$$

where ds_1=10, ds_2=5 and ds_3=7. Let $S_{r,p}$ be a subset of L_r such that $S_{r,p}$ = *{train project samples in* L_r *that have a number of cost factors* $NF \geq RT_j$ *whose values are equal to those of sample p}* which has $n_{r,p}$ elements. Then the enhanced effort estimation is calculated as:

$$\widehat{eff}_{r,p} = \frac{1}{n_{r,p}} \left(\sum_{i=1}^{n_{r,p}} eff_{i,r,p}\right) \pm \sigma_{n_{r,p}} \tag{12}$$

The Hit Ratio (*HR*) for the enhanced *RM* of the K_r test samples is defined in (13), where K_r = *{test project samples in* C_r *that satisfy inequality (15)}*.

$$HR_{RM,r} = \frac{|K_r|}{n_{testing,r}} \tag{13}$$

$$n_{RM,r} = |K_r| \tag{14}$$

$$\frac{1}{n_{train,r,p}} \left(\sum_{i=1}^{n_{r,p}} eff_{i,r,p}\right) - \sigma_{n_{train,r,p}} \leq \widehat{eff}_{r,p} \leq \frac{1}{n_{train,r,p}} \left(\sum_{i=1}^{n_{r,p}} eff_{i,r,p}\right) + \sigma_{n_{train,r,p}} \tag{15}$$

4 Empirical Experiments and Results

The FDTs were constructed with SPSS v.17.0 and additionally to CHAID and CART algorithms the exhaustive CHAID algorithm was also utilized to examine all possible splits for each predictor. For each algorithm different parameters were tested. We varied the minimum number of cases per parent and child node for both CHAID and CART and tried different splitting criteria. The maximum tree depth was confined to the maximum number of variables included in each cost Driver Scheme (DS) S1, S2 or S3. For CHAID the significance value was initially adjusted by the Bonferroni method to produce the simplest trees, but also varying values were examined to allow for more splitting nodes. Additionally, the Pearson chi-squared test statistic was used. For CHAID and CART the minimum change in improvement was set to 0.001 and 0.0001 respectively. For CART both the Twoing and Gini splitting methods were tested to either maximize the homogeneity of the child nodes with respect to the value of the target variable, or to create binary splits. Finally, cross-validation for both CHAID and CART used 10 sample folds. Table 3 lists a condensed set of the most significant and more frequently produced rules along with the higher *SL* and *NF* ratings obtained with each DS and tree algorithm. Finally, Table 4 lists the results of the classification process for the best 14 rules appearing in the trees at the validation phase.

The results from the experiments conducted are separately displayed in the training and testing phases. From the initial 673 train and 288 test project samples we calculate the mean effort and standard deviation of the respective *n* number of samples that satisfy each rule (displayed in every row of Table 4). The Significance Level (*SL)* obtained from the classified projects is consistently close or equal to the unit, indicating with high confidence that the classification performed was successful. For the same projects the small standard deviation reports that the range effort and mean effort prediction obtained is near to the actual effort value, indicating relatively good prediction accuracy. During testing the standard deviation values remained at similar levels as with the train samples which indicate generalization and stability of the approach when different samples are examined. Additionally, the best *HR* levels

Table 3. A list of indicative rules

DS	Algorithm	NF	SL	'If' Part of Rule and 'Then' Part of Rule
S1	CHAID	2	0.974	**IF** ((ATS != "HIGH" AND ATS != "MEDIUM") AND (PET != "MEDIUM" AND PET != "HIGH") AND (DT != "New Development")) **THEN** EFFORT="MEDIUM"
S2	CART	4	1.000	**IF** (RL != "R4" AND RL != "R3") AND (((DT = "Enhancement" OR DT = "Re-development") OR (DT != "New Development") AND ((DP = "" OR DP = "MF") OR (DP != "Multi" AND DP != "MR" AND DP != "PC") AND ((LT = "3GL" OR LT = "" OR LT = "4GL" OR LT = "2GL") OR (LT != "ApG") AND (RL != "R2"))))) AND (((LT = "4GL" OR LT = "ApG" OR LT = "2GL") OR (LT != "3GL" AND LT != "") AND (DP = "Multi"))) **THEN** EFFORT="MEDIUM"
S3	exCHAID	7	0.991	**IF** ((DT != "New Development") AND (DP = "MF" OR DP = "Multi") AND (AT!= "" AND AT!= "Stock control & order processing;" AND AT!= "Transaction/Production System;" AND AT!= "Maintenance;")) **THEN** EFFORT="MEDIUM"

Table 4. Indicative experimental results of rules validation

DS	Algorithm	SL	TRAINING			TESTING			PM		RM	
			n(673)	$\widehat{eff}$	σ_n	n(288)	$\widehat{eff}$	σ_n	HR	HR(%)	HR	HR(%)
S1	CHAID	0.939	155	7.332	1.075	66	7.099	1.194	42/66	63.63	42/66	63.64
S1	CHAID	0.974	418	7.118	1.279	168	6.985	1.383	108/168	64.28	107/168	63.69
S1	**CART**	**0.953**	**632**	**7.386**	**1.252**	**262**	**7.265**	**1.354**	**166/262**	**63.35**	**170/262**	**64.89**
S1	**CART**	**0.959**	**630**	**7.380**	**1.248**	**261**	**7.255**	**1.348**	**166/261**	**63.60**	**169/261**	**64.75**
S1	exCHAID	0.909	10	8.353	1.359	4	8.330	0.634	3/4	75.00	3/4	75.00
S1	**exCHAID**	**0.896**	**43**	**7.831**	**1.079**	**20**	**7.669**	**1.232**	**11/20**	**55.00**	**12/20**	**60.00**
S2	CHAID	0.971	269	7.008	1.373	106	6.942	1.489	68/106	64.15	68/106	64.15
S2	CHAID	0.969	405	7.109	1.298	153	6.899	1.396	97/153	63.39	99/153	64.71
S2	**CART**	**0.873**	**96**	**8.013**	**1.007**	**41**	**6.711**	**1.169**	**21/41**	**51.21**	**21/41**	**51.22**
S2	CART	1.000	92	6.766	1.101	33	5.217	1.143	22/33	66.66	21/33	63.64
S3	**CHAID**	**0.973**	**403**	**7.097**	**1.291**	**152**	**6.904**	**1.399**	**96/152**	**63.15**	**98/152**	**64.47**
S3	CHAID	0.921	210	7.931	1.009	91	7.749	1.155	52/91	57.14	51/91	56.04
S3	exCHAID	0.991	110	7.058	1.005	43	6.867	1.165	28/43	65.11	27/43	62.79
S3	**exCHAID**	**1.000**	**60**	**7.859**	**1.014**	**22**	**7.420**	**1.198**	**10/22**	**45.45**	**11/22**	**50.00**

obtained by the testing phase with respect to the *PM* and *RM* thresholds set earlier is equal to 75% and 67%. The best in accuracy prediction rule satisfies only a very small number of samples, we can conclude that the more homogeneous clusters of data are created the more prediction accuracy is improved.

Considering the two validation threshold measures used we would expect that the *RM* enhancement, since it adds restrictiveness to the rules, would have produced more accurate results, but this was not always the case. Even though the best *HR* yielded was again 75% followed by 65% of the cases, overall we observe that prediction was slightly improved in 6 rules (these are marked in bold in Table 4), remained the same in 4 whereas deteriorated in only 4 rules. This shows that validation may, of course, be further improved. Nevertheless, the consistent high level of accuracy prediction at the levels of 65-70% of the cases indicates that the approach may be considered promising

and provides a basis for further experimentation and improvement. Finally, the majority of the rules obtained in S1 where all the attributes participated, consistently placed similar cost drivers, namely DT, DP, ATS and PET, at the FDT top levels, indicating that these are the most decisive attributes describing or driving effort in the specific dataset. This may lead to infer that the approach is able to express complex relationships among the project attributes under investigation and effort and that within the enhanced classification clusters produced we may achieve adequately successful estimations of 65-70% accuracy in the range of values involved.

5 Conclusions

The creation of a model able to combine techniques and methods to offer reliable software effort estimations especially from the initial development phases is not a trivial task. This work attempted to address this problem and utilize a combination of techniques to build a data-driven model based on Fuzzy Decision Trees (FDT) enhanced with fuzzy logic. The idea was to produce stable and strong classification rules and use them to perform cost predictions. Each rule isolates a number of project samples which satisfy a number of cost parameters in terms of similar values, thus forming clusters of past project data. When a new project is classified in a cluster (i.e., satisfies the corresponding rule) then the estimated effort value is expressed in terms of the mean value and standard deviation of the group's effort.

The FDT were produced by the CHAID and CART algorithms, while their structure was further analyzed and enhanced to investigate the possibility of improving estimation accuracy by increasing homogeneity within the cluster. This was performed by introducing similarity measures that aimed at restricting further the classification of projects in a cluster by taking into consideration additional cost factors apart from those included in the corresponding rule. Thus, we would be able to lower discrepancy and diversification among the classified samples that were subsequently used as guidance in the prediction process. Three different data schemes were constructed so as to assess the performance of our approach on different sets of parameters and attempt also to produce a rough ordering of significance among the cost drivers involved. The first set included all available parameters, the second only those of categorical nature, while the third may be regarded as the most interesting and practical as it contained only cost parameters that may be measured during the early phases of development.

The experiments conducted yielded sufficiently accurate effort predictions for all data schemes tested, with 65% to 75% of the cases having the estimated effort close to the actual value, below a threshold level represented by the standard deviation of the project samples classified in the cluster in focus. The performance of the classification rules produced by the FDTs was found to be directly related to the tree construction algorithm and the parameters used for prediction and evaluation. In addition, it became evident through the experiments that the quality of the results is highly affected by the quality of the data, as well as the method of fuzzification. Finally, we may conclude that the FDT formed the basis to discover the interrelations among vital project variables and suggested that categorical cost factors are of high importance in determining the evolution of the final effort spent during development.

As such they must be given higher attention than what have been received until today and should be more regularly included in cost estimation models.

Our future work will continue with further experimentation with the rules obtained and development of new tree structures, as this is a tedious try-and-locate task. In this context we plan to integrate our tree construction process with optimization techniques, and more specifically with Genetic Algorithms, so as to try producing better trees and hence better rules by searching for combinations of cost factors (sub-schemes) as well as for the near to optimal settings of the tree creation algorithm. In this context it would be very interesting to investigate whether prediction success may be guided to higher levels by examining which factors may be taken into consideration for the enhanced part of the classification. More specifically, currently we are only concerned with having half or more of the factors that do not participate in the rule resemble those of the new project and not with which factors exactly present this similarity with the new project under estimation. We will attempt to control the type of factors in this resemblance evaluation process aiming at identifying those that are in favour of classifying the available data better (stricter). Finally, another research step we intend to take is to investigate a more 'sophisticated' mechanism of exploiting the sample values of the projects classified in a cluster, like regression techniques or nearest neighbours' algorithms, so as to reach to better local approximations.

References

1. Gray, A.R., MacDonell, S.G.: Applications of Fuzzy Logic to Software Metric Models for Development Effort Estimation. In: Proceedings of 1997: Annual Meeting of the North American Fuzzy Information Processing Society – NAFIPS, Syracuse NY, USA, pp. 394–399. IEEE, Los Alamitos (1997)
2. Boehm, B.W.: Software Engineering Economics. Prentice-Hall, Englewood Cliffs (1981)
3. Little, T.: Schedule estimation and uncertainty surrounding the cone of uncertainty. IEEE Software 23, 48–54 (2006)
4. Mair, C., Shepperd, M.: The consistency of empirical comparisons of regression and analogy-based software project cost prediction. In: International Symposium on Empirical Software Engineering, pp. 509–518 (2005)
5. Moløkken, K., Jørgensen, M.: A review of software surveys on software effort estimation. In: Proceedings of the International Symposium on Empirical Software Engineering, pp. 223–230. IEEE Computer Society, Los Alamitos (2003)
6. Jørgensen, M., Shepperd, M.: A Systematic Review of Software Development Cost Estimation Studies. Software Engineering. IEEE Transactions on Software Engineering 33(1), 33–53 (2007)
7. Gruschke, T.M., Jørgensen, M.: The role of outcome feedback in improving the uncertainty assessment of software development effort estimates. ACM Transactions of Software Engineering Methodology 17, 1–35 (2008)
8. Valerdi, R.: Cognitive Limits of Software Cost Estimation. In: First International Symposium on Empirical Software Engineering and Measurement, pp. 117–125 (2007)
9. Boehm, B.W., Abts, C., Brown, A., Chulani, S., Clark, B., Horowitz, E., Madachy, R., Reifer, D., Steece, B.: Software Cost Estimation with COCOMO II. Pearson Publishing, London (2000)

10. Albrecht, A.J.: Measuring Application Development Productivity. In: Proceedings of the Joint SHARE, GUIDE, and IBM Application Developments Symposium, pp. 83–92 (1979)
11. Putnam, L.H.: A General Empirical Solution to the Macro Software Sizing and Estimating Problem. IEEE Transactions on Software Engineering 4(4), 345–361 (1978)
12. Jun, E.S., Lee, J.K.: Quasi-optimal Case-selective Neural Network Model for Software Effort Estimation. In: Expert Systems with Applications, vol. 21(1), pp. 1–14. Elsevier, New York (2001)
13. Briand, L.C., Basili, V.R., Thomas, W.M.: A Pattern Recognition Approach for Software Engineering Data Analysis. IEEE Transactions on Software Engineering 18, 931–942 (1992)
14. Chatzoglou, P.D., Macaulay, L.A.: A Rule-Based Approach to Developing Software Development Prediction Models. Automated Software Engineering 5, 211–243 (1998)
15. Briand, L.C., Wust, J.: Modeling development effort in object-oriented systems using design properties. IEEE Transactions on Software Engineering 27, 963–986 (2001)
16. Burgess, C.J., Leftley, M.: Can Genetic Programming Improve Software Effort Estimation? A Comparative Evaluation. In: Information and Software Technology, vol. 43(14), pp. 863–873. Elsevier, Amsterdam (2001)
17. MacDonell, S.G., Shepperd, M.J.: Combining Techniques to Optimize Effort Predictions in Software Project Management. Journal of Systems and Software 66(2), 91–98 (2003)
18. Xu, Z., Khoshgoftaar, T.M.: Identification of Fuzzy Models of Software Cost Estimation. Fuzzy Sets and Systems 145(1), 141–163 (2004)
19. Huang, S.-J., Lin, C.-Y., Chiu, N.-H.: Fuzzy Decision Tree Approach for Embedding Risk Assessment Information into Software Cost Estimation Model. Software Engineering and Software 22, 297–313 (2006)
20. Andreou, A.S., Papatheocharous, E.: Software Cost Estimation using Fuzzy Decision Trees. Automated Software Engineering, 371–374 (2008)
21. Zadeh, L.A.: Fuzzy Set. Information and Control 8, 338–353 (1965)
22. Quinlan, J.R.: Induction of Decision Trees. Machine Learning 1, 81–106 (1986)
23. Kass, G.V.: An Exploratory Technique for Investigating Large Quantities of Categorical Data. Applied Statistics 20(2), 119–127 (1980)
24. Breiman, L., Friedman, J., Oshlen, R., Stone, C.: Classification and Regression Trees. Wadsworth International Group (1984)
25. International Software Benchmarking Standards Group (ISBSG), Estimating, Benchmarking & Research Suite Release 9, ISBSG, Victoria (2005), http://www.isbsg.org/
26. Braz, M.R., Vergilio, S.R.: Using Fuzzy Theory for Effort Estimation of Object-Oriented Software. In: Proceedings of the 16th IEEE International Conference on Tools with Artificial Intelligence, pp. 196–201 (2004)

s-OLAP: Approximate OLAP Query Evaluation on Very Large Data Warehouses via Dimensionality Reduction and Probabilistic Synopses

Alfredo Cuzzocrea

ICAR-CNR and University of Calabria, 87036 Cosenza, Italy
cuzzocrea@si.deis.unical.it

Abstract. In this paper, we propose *s*-OLAP, a framework for supporting approximate range query evaluation on data cubes that meaningfully makes use of two innovative perspectives of OLAP research, namely *dimensionality reduction* and *probabilistic synopses*. The application scenario of *s*-OLAP is a networked and heterogeneous very large Data Warehousing environment where applying traditional algorithms for processing OLAP queries is too much expensive and not convenient because of the size of data cubes, and the computational cost needed to access and process multidimensional data. *s*-OLAP relies on intelligent data representation and processing techniques, among which: (*i*) the amenity of exploiting the *Karhunen-Loeve Transform* (KLT) for obtaining dimensionality reduction of data cubes, and (*ii*) the definition of a probabilistic framework that allows us to provide a rigorous theoretical basis for ensuring *probabilistic guarantees* over the degree of approximation of the retrieved answers, which is a critical point in the context of approximate query answering techniques in OLAP.

Keywords: Data Cube Compression, Approximate Query Answering, OLAP.

1 Introduction

Multidimensional conceptual/data models represent data as univocally associated to unique positions in a multidimensional space, and support query and mining tasks over *data cubes* [11] stored in *Data Warehouse Servers* (DWSs) according to a multi-resolution view of data. The growing attention towards such models has been stirred up by recent advances of *On-Line Analytical Processing* (OLAP) systems [4], which allow us to efficiently support intelligent data analysis for a wide range of modern application scenarios ranging from *Business Intelligence* to *Sensor Network Data Analysis Tools*. Traditionally, OLAP technology has been proposed with the goal of supporting *just-in-time*, *summarized* knowledge extraction in decision-making processes of very large organizations. Despite this initial goal, the reliability and effectiveness of OLAP has made OLAP engines a (very) popular component of a plethora of *Data- and Knowledge-Intensive Systems*.

The data cube [11] is the fundamental conceptual/data model of OLAP. In OLAP scenarios, a data cube effectively supports the above-mentioned decision-making

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 248–262, 2009.
© Springer-Verlag Berlin Heidelberg 2009

analysis goals, thanks to some meaningfully abstractions of data and schemas kept in the relational data source. According to the data cube conceptual/data model, multidimensional data are organized in cubes characterized by a set of *dimensions* and a set of *measures*, which, originally, are attributes defined in the schema of the relational data source. Dimensions, which model the analysis parameters (e.g., intervals of time, regions, products), or, more properly, the *OLAP functional attributes*, allow us to univocally locate *data cells* storing measures, or, more properly, *OLAP measure attributes*, which are the values of interest for the target decision-making process. In turn, measures found on traditional SQL *aggregate operators* like COUNT, SUM, AVG etc over the set of referred relational tuples. For instance, a three-dimensional SUM-based data cube on sale data could store the total amount of sales (i.e., the measure) due to products (i.e., the first dimension) belonging to the class *"Electrics"* sold in the region (i.e., the second dimension) *"South Italy"* during the interval of time (i.e., the last dimension) *2000-2005*. Based on this multidimensional model, various kinds of OLAP queries retrieving data by means of multidimensional aggregations are executed against data cubes.

As regards storage issues, let $D = \{d_0, d_1, \ldots, d_{n-1}\}$ be the set of dimensions of a given data cube A, according to the popular MOLAP storage organization [12], A is represented in memory as an n-dimensional array. *Range queries* [15] are an important class of OLAP queries that very often are executed against data cubes. They are defined as the application of a given SQL aggregate operator over a set of selected contiguous ranges in the dimensional domains. For instance, an n-dimensional range-SUM query over an n-dimensional data cube A can be generally formulated as follows:

$$\mathsf{SUM}(x_0 : y_0, x_1 : y_1, \ldots, x_{n-1} : y_{n-1}) = \sum_{x_0 \le i_0 \le y_0} \sum_{x_1 \le i_1 \le y_1} \cdots \sum_{x_{n-1} \le i_{n-1} \le y_{n-1}} A[i_0][i_1]\ldots[i_{n-1}] \quad (1)$$

such that $\langle x_k{:}y_k \rangle$ denotes the range defined on the dimension d_k. In the rest of the paper, for the sake of simplicity we will assume of dealing with range-SUM queries, but extending synopsis data structures and techniques we propose as to deal with other classes of queries (e.g., range-COUNT, range-AVG etc) is straightforward.

Despite the complexity and the resource-intensiveness of processing range queries against very large data cubes stored in DWSs, client-side users/applications performing OLAP are very often characterized by small amount of memory, small computational capability, and customized tools with interactive, graphical user interface supporting *qualitative*, *trend* analysis only. This evidence makes computing *approximate answers* [2] to range queries more suitable and efficient than computing exact answers [5]. In fact, typical decision-support-queries can be very resource-intensive in terms of spatial and temporal computational needs, whereas an approximate evaluation of queries perfectly fits application requirements of OLAP [5]. Obviously, since this computation paradigm introduces approximation, *accuracy of answers* must also rigorously taken into consideration. Based on this theory, the *synopsis data structures* [2] proposal has appeared in literature. Synopses are succinct, *summarized representation* of massive data structures, like OLAP data cubes, which must fit in a given input storage space bound B, while minimizing the *query error* due to the introduced approximation. According to the approximate query

answering paradigm, OLAP queries are executed on synopses rather than the original data cube, thus speeding-up query response time.

All considering, proving *fast*, *exploratory* answers with some *guarantees* on their degree of approximation is a critical challenge for OLAP research. This challenge can be accomplished via pursuing the following goals: (*i*) minimizing the time complexity of OLAP query processing algorithms by *decreasing* the number of disk I/Os needed to access and process multidimensional data, and (*ii*) ensuring the quality of approximate answers with respect to the exact ones by providing some *guarantees* on the accuracy of the approximation.

With these ideas in mind, in this paper, we introduce *s*-OLAP, a framework for efficiently supporting approximate OLAP query evaluation on very large data warehouses via the meaningful metaphors of *dimensionality reduction* and *probabilistic synopses*. Particularly, the usage of probabilistic synopses allows us to ensure *probabilistic guarantees* on the retrieved approximate answers.

2 s-OLAP: An Overview

s-OLAP (see Figure 1) attempts at solving the above-described issues in dealing with querying very large data warehouses, provided by OLAP interfaces, and, on the basis of dimensionality reduction and probabilistic synopses, defines a methodology that allows us to obtain fast, approximate answers to range queries over multidimensional data cubes. *s*-OLAP is a multi-user framework that sits between a (very large) DWS and OLAP clients that interact with the former via performing query and mining tasks of business processes. In this respect, the main idea of *s*-OLAP relies on efficiently exploiting the *data cube compression/approximation paradigm* [2].

s-OLAP comes from the convergence of some approximate query answering techniques for OLAP, namely: [5], where the dimensionality reduction technique for multidimensional data cubes is proposed; [6], which describes a technique that can be reasonably considered as a generalization of the technique proposed in this paper; [8], where a similar probabilistic framework is proposed; [7], which firstly introduces an approach for dealing with the dynamics of OLAP queries.

The core of *s*-OLAP is the innovative dimensionality reduction technique [5]. In more detail, with respect to [5], in this paper (*i*) we efficiently exploit the theoretical results presented in [5] by also integrating them inside a real-life framework (i.e.,

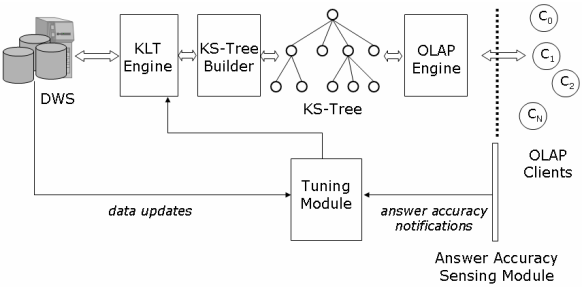


Fig. 1. The Framework *s*-OLAP

s-OLAP), and (*ii*) we introduce our probabilistic synopses for supporting approximate OLAP query answering that meaningfully match with the technique [5].

A very important issue in approximate query processing research consists in dealing with the multidimensionality of data domains and queries. In fact, many state-of-the-art techniques focused on the approximation of set-valued queries on relational databases, one-dimensional and two-dimensional queries on data cubes have been proposed during these last years, but, actually, there are few techniques allowing us to efficiently evaluate with approximation multidimensional queries on multidimensional data cubes. In order to address this critical challenge, in [5] a dimensionality reduction technique, based on the well known *Karhunen-Loeve Transform* (KLT) [18], is proposed. KLT allows us to obtain an m-dimensional data domain L_m from a given n-dimensional data domain L_n, such that $m << n$, such that L_m provides a *summarized description* of L_n. In more detail, in [5] a *MOLAP- based representation technique for data cubes* that allows us to significantly reduce the overall error caused by the projection of the input n-dimensional data cube over a smaller m-dimensional data domain is introduced, with the goal of applying the KLT to these MOLAP data cubes directly, instead than the original data cube, along with some other effective optimizations useful to "tailor" the KLT to efficiently support approximate query answering in OLAP.

In *s*-OLAP, the technique [5] is used to obtain a *collection of two-dimensional data domains* (i.e., $m = 2$), denoted by CoL_2, from the MOLAP data cubes representing the input n-dimensional data cube A, and *probabilistic synopses are computed for these two-dimensional data domains*. The overall process is bounded by the input storage space B available for housing the synopsis data structure. This process finally originates an R^+-tree like synopsis data structure for A, namely *KS-Tree*(A). *KS-Tree* is a persistent object that represents the knowledge kept in A in a summarized fashion, and provides efficient access and query evaluation over summarized data it stores. In more detail, each query Q against the original data cube A is (*i*) redirected to *KS-Tree*(A), and (*ii*) evaluated against *KS-Tree*(A), thus providing an approximate answer to Q, denoted by $\tilde{A}(Q)$. According to this approximation paradigm, the *relative query error* associated to Q, denoted by $\varepsilon(Q)$, is defined with respect to the exact answer to Q, $A(Q)$, which is evaluated against A, as follows:

$$\varepsilon(Q) = \frac{|A(Q) - \tilde{A}(Q)|}{A(Q)} \tag{2}$$

The underlying theoretical framework for providing probabilistic guarantees on the degree of approximation of the retrieved answers is another important feature of *s*-OLAP. To this end, we adopt the so-called *tail inequalities*, which are important results coming from the theoretical statistics. Such inequalities give probabilistic bounds on the estimation of a parameter of a given statistics (such as mean value, variance, co-variance etc) defined on a set of *random variables*. In theoretical statistics, a well-known, strong assumption claims that, by using a tail inequality in the estimation of a given observed parameter ρ, the error due to the estimate of ρ is at most ε with a probability that is at least $1 - \delta$, with $\varepsilon > 0$ and $\delta > 0$.

We highlight that this property plays a critical role in our research. In fact, providing approximate answers without any bounds on their accuracy would be fruitless. To

support this feature, *s*-OLAP builds probabilistic synopses starting from the transformed two-dimensional data domains in CoL_2 in such a way that these synopses "encode" the Hoeffding's inequality [16], and computes approximate answers against them. In other words, this approach permits us to build a *reliable estimator* for OLAP queries. In turn, this allows us to provide probabilistic bounds, or, in other terms, confidence intervals on the retrieved answers. The idea of using this kind of tail inequalities, first proposed in [14], has also been used in other approximate OLAP query answering papers, such as [6], which, similarly to the proposal of this paper, founds on the KLT decomposition [5], and applies the inequality to MOLAP cubes coming from the MOLAP-based data cube representation technique [5] but *having dimensionality higher than two*, and [8], which applies the inequality to *multidimensional buckets of tunable partitions of multidimensional data cubes able to deal with outliers stored in data cubes*. Therefore, the main difference between this paper and [6] is represented by the novelty of using two-dimensional data domains instead than more general MOLAP cubes. This approach has been suggested by [7], where a *meaningful decomposition of multidimensional data cubes into a set of two-dimensional data cubes by means of simple-yet-effective OLAP slice operations for approximate query answering purposes* is proposed. As highlighted in [7], the ratio of choosing the two-dimensional ones as output data domains of the KLT allows us to design a very efficient query strategy (see Section 4) that introduces low spatio-temporal overheads because of low-dimensional domains (i.e., $|D| = 2$) are processed.

KS-Tree is also a *self-adjusting* data structure. *s*-OLAP periodically re-configures the *KS-Tree* according to the *current accuracy* of the retrieved approximate answers, which is measured on the basis of an empirically-determined threshold τ, with $\tau > 0$. As we describe in Section 5, this useful feature is also the starting point for defining a *data exchanging protocol* between OLAP clients and *s*-OLAP, focused on "mediating" the degree of the approximation of answers provided by *s*-OLAP. The first proposal on this novel class of data warehouse tools appears in [7].

3 KS-Tree: A Synopsis Data Structure for OLAP

KS-Tree Builder (see Figure 1) is in charge of building the *KS-Tree* from the collection of two-dimensional data domains CoL_2. Let $L_{2,k}$ be the k-th domain belonging to CoL_2, such that k is in $\{0, 1, \ldots, |CoL_2| - 1\}$. According to our proposed approach, a (probabilistic) synopsis data structure, denoted by $L^P_{2,k}$, is obtained from the domain $L_{2,k}$ by means of the Hoeffding's inequality [16] (Section 3.1). As a result, $L^P_{2,k}$ is a synopsis data structure for $L_{2,k}$. By iterating this task for each domain $L_{2,k}$ belonging to CoL_2, a collection of synopsis data structures, denoted by CoL^P_2, is obtained. CoL^P_2 plus the tree-like indexing data structure forms the overall synopsis data structure for A, *KS-Tree*(A).

Please note that, if your email address is given in your paper, it will also be included in the meta data of the online version.

3.1 Building and Querying Probabilistic Synopses

Given a two-dimensional domain $L_{2,k}$ belonging to the collection CoL_2, we make use of the Hoeffding's inequality [16] for building the synopsis data structure $L^P_{2,k}$

belonging to the collection CoL^P_2. Without any loss of generality, we can assume that, similarly to $L_{2,k}$, even $L^P_{2,k}$ has a two-dimensional nature. Note that several very efficient solutions for representing-in-memory two-dimensional data structures are available.

The Hoeffding's inequality asserts the following. Let (*i*) $Z = \{Z_0, Z_1, ..., Z_{M-1}\}$ be a set of *independent random variables*, (*ii*) r a scalar such that $0 \leq Z_m \leq r$ with $m \in \{0, 1, ..., M-1\}$, (*iii*) $\overline{Z} = \frac{1}{M} \cdot \sum_{m=0}^{M-1} Z_m$ the sample mean of Z, (*iv*) μ the average value of Z. Then, for each $\varepsilon > 0$, the following inequality holds:

$$P\left(\left|\overline{Z} - \mu\right| \leq \varepsilon\right) \geq 1 - 2 \cdot e^{-\frac{2 \cdot M \cdot \varepsilon^2}{r^2}} \tag{3}$$

i.e. a *probabilistic bound* for the event $\mu = \overline{Z} \pm \varepsilon$, or, from the database perspective, a probabilistic bound for the (approximate) answer to the (one-dimensional) OLAP query $Q = \mathsf{AVG}(\Delta\text{-}r{:}\Delta\text{+}r)$, with $\Delta > 0$, is derived.

In order to exploit the Hoeffding's inequality in *s*-OLAP, a random variable set must be defined on the target data domain $L_{2,k}$. To this end, given a two-dimensional domain $L_{2,k}$ belonging to the collection CoL_2, we introduce two random variables, denoted by $Z_{X,k}(L_{2,k})$ and $Z_{Y,k}(L_{2,k})$, respectively, and defined as follows:

$$\begin{aligned} Z_{X,k} &: Dom(d_X(L_{2,k})) \rightarrow Dom(d_X(L_{2,k})) \\ Z_{Y,k} &: Dom(d_Y(L_{2,k})) \rightarrow Dom(d_Y(L_{2,k})) \end{aligned} \tag{4}$$

such that $Dom(d_j(L_{2,k}))$ denotes the domain of the dimension $d_j(L_{2,k})$ of $L_{2,k}$, with $j \in \{X, Y\}$. In such a way, a *multivariate stochastic system* $\Psi\{L_{2,k}\} = \{Z_{X,k}(L_{2,k}), Z_{Y,k}(L_{2,k})\}$ is obtained for $L_{2,k}$. Each random variable $Z_{j,k}$ in $\Psi\{L_{2,k}\}$, with $j \in \{X, Y\}$, performs a *random sampling* over the corresponding domain $Dom(d_j(L_{2,k}))$, and returns an indexer in $Dom(d_j(L_{2,k}))$, denoted by $\hat{Z}_{j,k}$, which represents an *instance* of $Z_{j,k}$.

The synopsis data structure $L^P_{2,k}$ is obtained from $L_{2,k}$ as an *instance* of $\Psi\{L_{2,k}\}$, denoted by $\hat{\Psi}\{L_{2,k}\}$, via obtaining two separate instances of the random variables $\hat{Z}_{j,k}$, i.e. $\hat{Z}_{X,k}$ and $\hat{Z}_{Y,k}$, respectively, i.e. $\hat{\Psi}\{L_{2,k}\} = \{\hat{Z}_{X,k}, \hat{Z}_{Y,k}\}$, and, as a consequence, the sampled data cell to be stored within $L^P_{2,k}$ is finally given by: $A[\hat{Z}_{X,k}][\hat{Z}_{Y,k}]$. The effective number of sampled data cells that is finally obtained is bounded by a *portion* of the available storage space B.

At query time, $\hat{\Psi}\{L_{2,k}\}$ allows us to obtain probabilistic guarantees over the degree of approximation of answers evaluated against $L^P_{2,k}$. Note that the random sampling-based generating approach for $\hat{\Psi}\{L_{2,k}\}$ ensures the needed feature of independence about the set of random variables [16]. Given a two-dimensional OLAP query $Q = \mathsf{SUM}(x_0{:}y_0, x_1{:}y_1)$ over $L_{2,k}$, the approximate answer to Q, $\tilde{A}(Q)$, can be obtained as follows:

$$\tilde{A}(Q) = |\hat{\Psi}\{L_{2,k}\}| \cdot \hat{\Psi}\{L_{2,k}\} \pm \varepsilon(Q) \tag{5}$$

being (*i*) $|\hat{\Psi}\{L_{2,k}\}|$ the cardinality of $\hat{\Psi}\{L_{2,k}\}$, and (*ii*) $\varepsilon(Q)$ the query error that is probabilistically bounded by (3). On a practical plane, the theoretical formula (5) is implemented as follows:

$$\tilde{A}(Q) = \sum_{x_0 \leq i_0 \leq y_0} \sum_{x_1 \leq i_1 \leq y_1} \underset{NULL}{L^P_{2,k}[i_0][i_1] \neq} L^P_{2,k}[i_0][i_1] \tag{6}$$

i.e. by accessing the sampled data cells stored in the region $R_Q = \langle [x_0{:}y_0], [x_1{:}y_1] \rangle$ contained by $L^P_{2,k}$, and discarding the null data cells. It should be noted that the latter approximate evaluation scheme involves much less computational overheads than the exact evaluation scheme (1).

3.2 KS-Tree Data Engineering Overview

From a data engineering perspective, *KS-Tree* is an R^+-tree like hierarchical data structure characterized by the following organization. Given a multidimensional data cube A, each leaf node n_k in *KS-Tree*(A) stores: (*i*) the multidimensional region R_k associated to the corresponding MOLAP data cube M_k originated by the MOLAP-based representation of A [5]; (*ii*) the two-dimensional synopsis data structure $L^P_{2,k}$ built from the two-dimensional data domain $L_{2,k}$ obtained from M_k via the KLT; (*iii*) the sum of *all* the sampled data cells contained by $L^P_{2,k}$, denoted by S_k; (*iv*) the *current* query error ε_k associated to $L^P_{2,k}$, being ε_k computed as the *average value* of the query errors due to *all* the synthetic queries that can be defined on $L^P_{2,k}$, and having a *selectivity* greater than a tuneable input parameter ξ_k (from (3), recall that this error is probabilistically bounded). More formally, let $QP(L^P_{2,k},\xi_k)$ denote the set of synthetic queries having selectivity greater than ξ_k that can be defined on $L^P_{2,k}$, ε_k is defined as follows:

$$\varepsilon_k = \frac{1}{|QP(L^P_{2,k},\xi_k)|} \cdot \sum_{j=0}^{|QP(L^P_{2,k},\xi_k)|-1} \varepsilon(q_j) \tag{7}$$

where q_j models a generic query belonging to $QP(L^P_{2,k},\xi_k)$.

Each internal node n_i in *KS-Tree*(A) stores: (*i*) a multidimensional region R_i that hierarchically contains the regions stored in *all* the nodes of the *KS-Tree*(A) sub-tree rooted in n_i, denoted by T_i; (*ii*) the sum of *all* the sampled data cells contained by *all* the two-dimensional synopsis data structures stored in *all* the *leaf* nodes of the *KS-Tree*(A) sub-tree rooted in n_i, T_i; denoted by S_i; (*iii*) the *current* query error ε_i associated to R_i, being ε_i computed as the *maximum error* among the query errors ε_j stored in *all* the nodes of the *KS-Tree*(A) sub-tree rooted in n_i, T_i. More formally, ε_j is defined as follows:

$$\varepsilon_i = \max_{n_j \in T_i} \{\varepsilon_j\} \tag{8}$$

How the regions of internal nodes of *KS-Tree*(A) are obtained? Starting from the multidimensional regions associated to the MOLAP data cubes originated by the MOLAP-based representation of the multidimensional data cube A, which are stored in the leaf nodes of *KS-Tree*(A), and given two input integer parameters P and ℓ, with $P > 0$ and $\ell > 0$, (*i*) parent nodes of leaf nodes store multidimensional regions that

hierarchically contain the multidimensional regions stored in a cluster of P leaf nodes, and (*ii*), according to a bottom-up approach, any other level of *KS-Tree*(A) adopts the same aggregation mechanisms with clusters of nodes having width equal to $P'' = \left\lfloor \frac{P'}{\ell} \right\rfloor$, such that P' is the width of the cluster associated to the lower level. The task above is iterated until the singleton multidimensional region corresponding to the region of the whole data cube A is achieved (the latter region is stored in the root node of *KS-Tree*(A)).

In more detail, given p n-dimensional regions belonging to a certain cluster v, $\{R_0, R_1, \ldots, R_{p-1}\}$, such that each region R_z in v is modeled as: $R_z = \langle [r_{l,z,0}:r_{u,z,0}], [r_{l,z,1}:r_{u,z,1}], \ldots, [r_{l,z,n-1}:r_{u,z,n-1}] \rangle$, where $r_{l,z,k}$ and $r_{u,z,k}$ denote the lower and the upper bound on the dimension d_k of R_z, respectively, the bounds of each range $[r_{l,v,j}:r_{u,v,j}]$ of the parent region $R_v = \langle [r_{l,v,0}:r_{u,v,0}], [r_{l,v,1}:r_{u,v,1}], \ldots, [r_{l,v,n-1}:r_{u,v,n-1}] \rangle$ are obtained according to the following formulas:

$$r_{l,v,j} = \bigcup_{z=0}^{p-1} r_{l,z,j} \quad r_{u,v,j} = \bigcup_{z=0}^{p-1} r_{u,z,j} \tag{9}$$

It should be noted that, being P and ℓ completely-free parameters, the overall process generating *KS-Tree*(A) can be easily tuned according to specific application requirements.

Let us to highlight some nice properties of *KS-Tree*, which, in turn, depend on the fact that *KS-Tree* is based on the R^+-tree. (*i*) R^+-tree is dynamic, thus making *KS-Tree* perfectly suitable to the goal of approximate query answering in OLAP with dynamic features (see Section 5). (*ii*) R^+-tree introduces low complexity, i.e. the computational cost required for accessing and managing multidimensional data is low. (*iii*) R^+-tree implicitly supports data aggregation as it is built over non-overlapping regions – this nice feature makes *KS-Tree* perfectly suitable to handle OLAP data. (*iv*) R^+-tree is paginated, so that the number of disk I/Os needed for accessing massive multidimensional data scales well, thus perfectly marrying with the goals of *KS-Tree*.

4 s-OLAP Query Model

In *s*-OLAP, an input multidimensional OLAP query Q against the target data cube A is evaluated against the synopsis *KS-Tree*(A), yet introducing some approximation. In particular, the approximate query evaluation scheme works differently in the dependency on the nature of Q.

If the ranges of Q cover exactly the ranges of a multidimensional region R_i associated to an internal node n_i of *KS-Tree*(A) (see Figure 2 (A)), then the approximate answer to Q, $\tilde{A}(Q)$, is directly retrieved as the value S_i with a (probabilistically-bounded) query error equal to ε_i, being S_i and ε_i both stored in n_i. Therefore, let $TN(Q)$ be the set of V internal *KS-Tree*(A) nodes involved by Q, denoted by:

$$TN(Q) = \{n_{Q,0}, n_{Q,1}, \ldots, n_{Q,V-1}\} \tag{10}$$

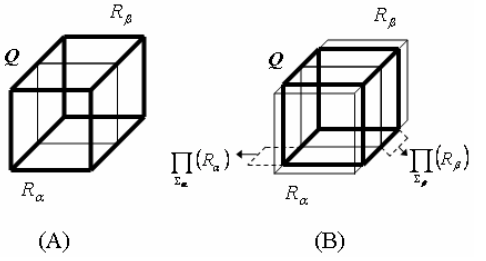


Fig. 2. Approximate Query Answering in s-OLAP

such that the ranges of Q cover exactly the ranges of regions of nodes in $TN(Q)$, $\tilde{A}(Q)$ is obtained as follows:

$$\tilde{A}(Q) = \sum_{v=0}^{V-1} S_v \tag{11}$$

where S_v denotes the sum value associated to the node n_v (see Section 3.2) It should be noted that this query scenario gives us the best setting with respect to the minimization of the query response time due to the evaluation of Q, as, in order to compute $\tilde{A}(Q)$, we do not need to reach the leaf level of *KS-Tree*(A).

If the ranges of Q does not cover exactly the ranges of a multidimensional region R_i associated to an internal node n_i of *KS-Tree*(A) (see Figure 2 (B)), then, in order to compute $\tilde{A}(Q)$, the leaf level of *KS-Tree*(A) must be reached. In this case, *Q is decomposed by the OLAP Engine* (see Figure 1) *in a set of W two-dimensional queries obtained with respect to the multidimensional regions R_k associated to a set of leaf nodes of KS-Tree*(A), denoted by:

$$TD(Q) = \left\{q^2_{Q,0}, q^2_{Q,1}, \ldots, q^2_{Q,W-1}\right\} \tag{12}$$

The decomposition of Q in two-dimensional queries is simply obtained by projecting the ranges of Q over the ranges of the (involved) W two-dimensional synopsis data structures $L^P_{2,k}$ stored in the *KS-Tree*(A) leaf nodes involved by Q, denoted by:

$$TS(Q) = \{L^P_{2,Q,0}, L^P_{2,Q,1}, \ldots, L^P_{2,Q,W-1}\} \tag{13}$$

The approximate answer to Q, $\tilde{A}(Q)$, is computed by summing-up all the approximate answers to the two-dimensional queries q^2_w in $TD(Q)$, as follows:

$$\tilde{A}(Q) = \sum_{w=0}^{W-1} \tilde{A}(q^2_{Q,w}) \tag{14}$$

The query error due to the approximate evaluation of Q, $\varepsilon(Q)$, is given by the following formula:

$$\varepsilon(Q) = \sum_{w=0}^{W-1} \varepsilon(q^2_{Q,w}) \tag{15}$$

being $\varepsilon(q^2_{Q,w})$ the query error due to the approximate evaluation of the two-dimensional query $q^2_{Q,w}$. From (6), (14) can be re-written as follows:

$$\tilde{A}(Q) = \sum_{w=0}^{W-1} \sum_{x_{0,w} \le i_{0,w} \le y_{0,w}} \sum_{x_{1,w} \le i_{1,w} \le y_{1,w}} \sum_{\substack{L^P_{2,Q,w}[i_{0,w}][i_{1,w}] \neq \\ NULL}} L^P_{2,Q,w}[i_{0,w}][i_{1,w}] \tag{16}$$

such that $\langle [x_{0,w}:y_{0,w}], [x_{1,w}:y_{1,w}] \rangle$ denotes the range of the query $q^2_{Q,w}$ over the two-dimensional synopsis data structure $L^P_{2,Q,w}$.

What about the probabilistic bound to (15)? Recall that each query error $\varepsilon(q^2_{Q,w})$ due to the approximate evaluation of $q^2_{Q,w}$ is bounded by (3). Since range-SUM queries are handled, let $\varepsilon(q^2_{Q,w^*})$ denote the *biggest one* among the W query errors due to the evaluation of the two-dimensional query q^2_{Q,w^*} in $TD(Q)$ over the two-dimensional synopsis data structure L^P_{2,Q,w^*} in $TS(Q)$, then the following inequality holds:

$$\varepsilon(Q) \le \varepsilon(q^2_{Q,w^*}) \tag{17}$$

such that:

$$P\left(\frac{1}{\| q^2_{Q,w^*} \|} \cdot \left| \tilde{A}(q^2_{Q,w^*}) - A(q^2_{Q,w^*}) \right| \le \varepsilon(q^2_{Q,w^*}) \right) \ge 1 - 2 \cdot e^{-\frac{4 \cdot \varepsilon^2(q^2_{Q,w^*})}{\max\{y_{0,w^*}, y_{1,w^*}\}^2}} \tag{18}$$

where $\| q^2_{Q,w} \|$ denotes the selectivity of $q^2_{Q,w}$ (due to (4), $M = 2$ in (18)). From (17) and (18), it follows the formula modeling the probabilistic bound for a given OLAP query Q involving $|TS(Q)|$ two-dimensional synopsis data structures $L^P_{2,k}$ stored in leaf nodes of *KS-Tree*(A):

$$P\left(\frac{1}{\| Q \|} \cdot \left| \tilde{A}(Q) - A(Q) \right| \le \varepsilon(Q) \right) \ge 1 - 2 \cdot e^{-\frac{4 \cdot |TS(Q)| \cdot \varepsilon^2(Q)}{\max\{y_{0,0}, y_{0,1}, \ldots, y_{0,W-1}, y_{1,0}, y_{1,1}, \ldots, y_{1,W-1}\}^2}} \tag{19}$$

Finally, note that, beyond the approximation due to sampling, $\varepsilon(Q)$ is also affected by the "intrinsic" approximation error due to the KLT, however mitigated by the MOLAP-based data cube representation technique (see Section 2).

5 Capturing and Handling the Dynamics of OLAP Queries

In [7], the concept of *Quality of Answer* (QoA) tools for DWS/OLAP is firstly introduced. Briefly, according to the philosophy of these tools, the target DWS and the OLAP clients to mediate on the degree of approximation of the retrieved answers, on the basis of a sort of *contract* between them. *s*-OLAP embeds several amenities making it able to efficiently support QoA tools, mainly thanks to its probabilistic framework. To this end, we express the accuracy α_Q required by an OLAP client for a

given (OLAP) query Q against A *as a percentage value* modeling the "distance" between the exact answer and the approximate answer (recall that the exact answer is *a-priori unknown* to the client). Therefore, let ε_Q the current query error associated to the *KS-Tree* node involved by Q, n_Q (internal or leaf node – see Section 3.2), in order to mediate on the degree of approximation of $\tilde{A}(Q)$, α_Q must be compared with the current query error ε_Q, which can be reasonably intended as the *maximum error* currently introduced by the *KS-Tree* for *that* query Q. In *s*-OLAP, ε_Q is thus a reliable metrics on which α_Q, for any OLAP query Q, can be measured, as ε_Q efficiently summarizes the "overall" error due to the *current* configuration of the *KS-Tree*. It should be noted that the realization provided by ε_Q implements the model based on the threshold τ introduced in Section 2.

When an OLAP client C_i issues an (OLAP) query Q against the target data cube A, the *OLAP Engine* (see Figure 1) intercepts Q, and performs a QoA protocol between *s*-OLAP and C_i, which is based on the following rules: (*i*) if $\alpha_Q \leq \varepsilon_Q$, then the accuracy required by C_i is supported by the current configuration of *KS-Tree* – the approximate answer $\tilde{A}(Q)$ is retrieved accordingly and forwarded to C_i, and no further actions concerning the configuration of *KS-Tree* are taken; (*ii*) if $\alpha_Q > \varepsilon_Q$, then the accuracy required by C_i is not supported by the current configuration of *KS-Tree* – similarly to the previous case, the approximate answer $\tilde{A}(Q)$ is retrieved and forwarded to C_i, with also embedding the degree of approximation, and, most importantly, the *Tuning Module* (see Figure 1) notifies the *KLT Engine* (see Figure 1) about the need of computing a more accurate *KS-Tree*, with benefits on the accuracy of "next" answers.

6 Experimental Assessment

In order to test the effectiveness of the approximate OLAP query answering technique we propose, we engineered a detailed experimental assessment, where we considered synthetic data cubes. Synthetic data cubes are indeed perfectly suitable to test the effectiveness of any OLAP data processing algorithm, as their free (experimental) input parameters used to determine the characteristics of final data distributions can be flexibly set in order to capture one or more aspects of interest to be observed during the experimental assessment (e.g., query performance, scalability, sensitivity with respect to some particular parameters etc). Obviously, this kind of analysis, which is extremely useful, must be completed by the experimental analysis on real-life data cubes, which is outside the scope of this paper, and as a consequence, left as future work.

As regards the data layer of our experimental assessment, we introduced the following synthetic data cubes: (*i*) **SyntUnif6D**, which is a six-dimensional MOLAP data cube storing Uniform data having *sparseness coefficient* $s = 0.005$, and occupying around 0.60 GB disk space; (*ii*) **SyntZipf6D**, which is a six-dimensional MOLAP data cube storing Zipfian data with $s = 0.004$, and occupying around 0.75 GB disk space. As regards experimental settings, we set the storage space B available to house the *KS-Tree* as equal to the 10 % of the whole synthetic data cube size. This

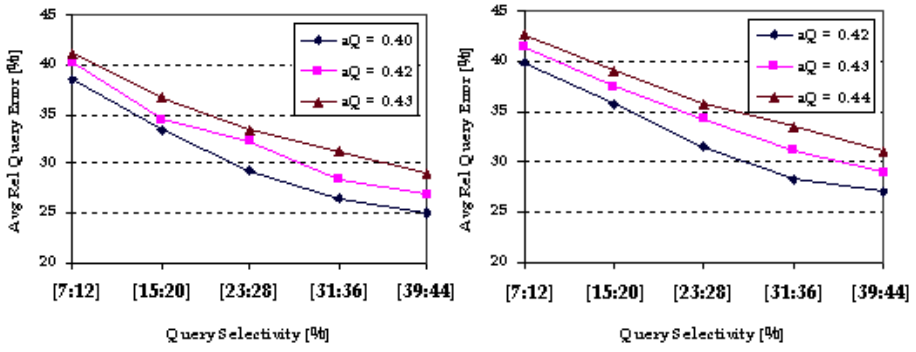


Fig. 3. Experimental Results on **SyntUnif6D** (*a*), and **SyntZipf6D** (*b*)

is a widely-accepted reference threshold for approximate query answering techniques in OLAP, as confirmed by a number of similar research efforts appeared in literature (e.g., [6,7,8]). As regards the query layer of our experimental assessment, given two input parameters $\Delta\varphi_0$ and $\Delta\varphi_1$, with $\Delta\varphi_0 > 0$ and $\Delta\varphi_1 > 0$, we considered random populations of queries having selectivity ranging in the interval $[\Delta\varphi_0{:}\Delta\varphi_1]$ % of the whole data cube volume. The ratio used to generate random populations of queries was that of considering a number of queries sufficient to "cover" the target synthetic data cube for more that 95 % of its volume. This ensured has a reliable experimental setting. Finally, as regards the metrics of our experimental assessment, we considered the *average relative query error* derived from (2). Let $QT_{\Delta\varphi_0,\Delta\varphi_1}(A)$ be the set of J (synthetic) queries for the target synthetic data cube A, denoted by:

$$QT_{\Delta\varphi_0,\Delta\varphi_1}(A) = \{q_{\Delta\varphi_0,\Delta\varphi_1,0}, q_{\Delta\varphi_0,\Delta\varphi_1,1}, \ldots, q_{\Delta\varphi_0,\Delta\varphi_1,J-1}\} \tag{20}$$

the average relative query error for queries $q_{\Delta\varphi_0,\Delta\varphi_1,j}$ in $QT_{\Delta\varphi_0,\Delta\varphi_1}(A)$, denoted by $\bar{\varepsilon}_{\Delta\varphi_0,\Delta\varphi_1}(A)$, is defined as follows:

$$\bar{\varepsilon}_{\Delta\varphi_0,\Delta\varphi_1}(A) = \frac{1}{|QT_{\Delta\varphi_0,\Delta\varphi_1}(A)|} \cdot \sum_{j=0}^{|QT_{\Delta\varphi_0,\Delta\varphi_1}(A)|-1} \varepsilon(q_{\Delta\varphi_0,\Delta\varphi_1,j}) \tag{21}$$

Figure 3 shows our experimental results on the synthetic data cubes **SyntUnif6D** (*a*) and **SyntZipf6D** (*b*), respectively, and for several values of the accuracy α_Q required by OLAP clients (see Section 5). As shown in Figure 3, the query error due to the approximate OLAP query answering technique we propose has a satisfactory trend that is in compliance with state-of-the-art OLAP research results of the literature (e.g., [6,7,8]). This evidence further confirms us the quality of the technique we propose. Moreover, it should be noted that, in our experiments, the intelligent data representation and processing techniques embedded in *s*-OLAP have always been able to meet the requirements posed by the accuracy threshold. Indeed, this nice dependency on an input accuracy threshold, which can be easily customized,

represents a novelty in approximate OLAP query answering research, and, for this reason, it could be reasonable considered as a point of innovation in this field.

Finally, it should be noted that our technique exposes a better query performance when Uniform data are processed rather than Zipfian data. This is due to the fact that, as widely-known, random sampling works better in approximating Uniform distributions rather than Zipfian ones (e.g., see [8]), so that the introduced query error is lower accordingly.

7 Related Work

Synopsis data structures have become very popular in Data Warehousing research, and, with more emphasis, in OLAP research. Given an input data domain L, a synopsis built on L, denoted by $s(L)$, is an ad-hoc data structure, often based on statistical or analytical properties of data stored in L, that is a summarized representation of L. $s(L)$ is used to evaluate queries against L with some approximation, often irrelevant for OLAP scenarios (e.g., see [5]), at the benefit of a lower query response time. During past years, many techniques focused to develop a methodology for building synopses, also in a semi-automatic manner, have been proposed [2]. From a theoretical point of view, almost all the proposed-in-literature approximate query answering techniques can be reasonably intended as techniques for building synopses, as their goal consists in building a succinct, compact representation of the input data domain, and using it at query time to speed-up query evaluation. Among all, we recall: *histograms* (e.g., [10,17,19,20,21]), *wavelets* (e.g., [3,22]), and *sampling* (e.g., [1,9]).

An histogram is a summarized representation of an input data domain L in terms of collection of buckets (i.e., data blocks storing some aggregate information about the range they refer), also called as *bucket partition*, and is used to obtain a reliable compressed representation of L useful to approximate query answering purposes. Many criteria to compute histograms have been proposed in literature. Among those, we recall: (*i*) minimizing the *skeweness* (i.e., the asymmetry) of data distributions of final buckets, (*ii*) minimizing the query error due to the approximate evaluation of a given family of input queries, (*iii*) satisfying a maximal query error threshold for arbitrary input queries, (*iv*) optimizing the sensitivity of the bucket partition to changing-in-nature query-workloads, and so forth. Wavelets are mathematical transformations that pursue to compress an input data domain L by means of a *hierarchical decomposition* of L in terms of the so-called *wavelet coefficients* that weight the so-called *wavelet basis functions*. The advantage of wavelets over histograms is represented by a greater flexibility at a cost of a lower capability of dealing with high-dimensional data domains. Sampling represents the actual trend for obtaining compression of massive data domains. The idea of sampling consists in generating a *sample data domain L'* from the input data domain L, according to a given distribution (e.g., Uniform). Sampling is particularly useful to compress high-dimensional data domains at a low spatio-temporal computational cost, with also appreciable flexibility features.

8 Conclusions and Future Work

In this paper, we have presented *s*-OLAP, a framework for supporting approximate OLAP query evaluation on very large data warehouses that relies on some effective intelligent data representation and processing techniques, namely the dimensionality reduction of multidimensional data cube and probabilistic synopses The core synopsis data structure of *s*-OLAP is the *KS-Tree*, a self-adjusting synopsis for OLAP, which has clearly demonstrated its benefits in the context of approximate query answering over data cubes. Future work is mainly focused on making *s*-OLAP able to support more complex OLAP aggregations, like those discussed in [13], rather than simple SQL-based aggregations (e.g., those based on COUNT, SUM, AVG etc). This will require the study of novel and more complex probabilistic estimators coming from theoretical statistics, and their meaningful adaptation to approximate query answering in OLAP as well.

References

1. Acharya, S., Gibbons, P.B., Poosala, V., Ramaswamy, S.: Join Synopses for Approximate Query Answering. In: Proc. of 1999 ACM SIGMOD Int. Conf., pp. 275–286 (1999)
2. Barbarà, D., Du Mouchel, W., Faloutsos, C., Haas, P.J., Hellerstein, J.M., Ioannidis, Y.E., Jagadish, H.V., Johnson, T., Ng, R.T., Poosala, V., Ross, K.A., Sevcik, K.C.: The New Jersey Data Reduction Report. IEEE Data Engineering Bulletin 20(4), 3–45 (1997)
3. Chakrabarti, K., Garofalakis, M., Rastogi, R., Shim, K.: Approximate Query Processing Using Wavelets. Very Large Data Bases Journal 10(2-3), 199–223 (2001)
4. Chaudhuri, S., Dayal, U.: An Overview of Data Warehousing and OLAP Technology. ACM SIGMOD Record 26(1), 65–74 (1997)
5. Cuzzocrea, A.: Overcoming Limitations of Approximate Query Answering in OLAP. In: Proc. of 9th IEEE IDEAS Int. Conf., pp. 200–209 (2005)
6. Cuzzocrea, A.: Providing Probabilistically-Bounded Approximate Answers to Non-Holistic Aggregate Range Queries in OLAP. In: Proc. of 8th ACM DOLAP Int. Works, pp. 97–106 (2005)
7. Cuzzocrea, A.: Accuracy Control in Compressed Multidimensional Data Cubes for Quality of Answer-based OLAP Tools. In: Proc. of 18th IEEE SSDBM Int. Conf., pp. 301–310 (2006)
8. Cuzzocrea, A., Wang, W.: Approximate Range-Sum Query Answering on Data Cubes with Probabilistic Guarantees. Journal of Intelligent Information Systems 28(2), 161–197 (2007)
9. Gibbons, P.B., Matias, Y.: New Sampling-Based Summary Statistics for Improving Approximate Query Answers. In: Proc. of 1998 ACM SIGMOD Int. Conf, pp. 331–342 (1998)
10. Gibbons, P.B., Matias, Y., Poosala, V.: Fast Incremental Maintenance of Approximate Histograms. ACM Transactions on Database Systems 27(3), 261–298 (2002)
11. Gray, J., Chaudhuri, S., Bosworth, A., Layman, A., Reichart, D., Venkatrao, M., Pellow, F., Pirahesh, H.: Data Cube: A Relational Aggregation Operator Generalizing Group-by, Cross-Tab, and Sub Totals. Data Mining and Knowledge Discovery 1(1), 29–53 (1997)
12. Han, J., Kamber, M.: Data Mining: Concepts and Techniques. Morgan Kauffmann Publishers, San Francisco (2000)

13. Han, J., Pei, J., Dong, G., Wang, K.: Efficient Computation of Iceberg Cubes with Complex Measures. In: Proc. of 2001 ACM SIGMOD Int. Conf., pp. 1–12 (2001)
14. Hellerstein, J.M., Haas, P.J., Wang, H.J.: Online Aggregation. In: Proc. of 1997 ACM SIGMOD Int. Conf., pp. 171–182 (1997)
15. Ho, C.-T., Agrawal, R., Megiddo, N., Srikant, R.: Range Queries in OLAP Data Cubes. In: Proc. of 1997 ACM SIGMOD Int. Conf., pp. 73–88 (1997)
16. Hoeffding, W.: Probability Inequalities for Sums of Bounded Random Variables. Journal of the American Statistical Association 58(301), 13–30 (1963)
17. Ioannidis, Y.E., Poosala, V.: Histogram-Based Approximation of Set-Valued Query Answers. In: Proc. of 25th VLDB Int. Conf., pp. 174–185 (1999)
18. Jain, A.K.: Fundamentals of Digital Image Processing. Prentice-Hall, Upper Saddle River (1989)
19. Poosala, V., Ganti, V.: Fast Approximate Answers to Aggregate Queries on a Data Cube. In: Proc. of IEEE 11th SSDBM Int. Conf., pp. 24–33 (1999)
20. Poosala, V., Ganti, V., Ioannidis, Y.E.: Approximate Query Answering using Histograms. IEEE Data Engineering Bulletin 22(4), 5–14 (1999)
21. Poosala, V., Ioannidis, Y.E.: Selectivity Estimation Without the Attribute Value Independence Assumption. In: Proc. of 23rd VLDB Int. Conf., pp. 486–495 (1997)
22. Vitter, J.S., Wang, M., Iyer, B.: Data Cube Approximation and Histograms via Wavelets. In: Proc. of 7th ACM CIKM Int. Conf., pp. 96–104 (1998)

Part II

Artificial Intelligence and Decision Support Systems

A Self-learning System for Object Categorization

Danil V. Prokhorov

Toyota Research Institute NA, TTC-TEMA, Ann Arbor, MI 48105, USA
dvprokhorov@gmail.com

Abstract. We propose a learning system for object categorization which utilizes information from multiple sensors. The system learns not only prior to its deployment in a supervised mode but also in a self-learning mode. A competition based neural network learning algorithm is used to distinguish between representations of different categories. We illustrate the system application on an example of image categorization. A radar guides a selection of candidate images provided by the camera for subsequent analysis by our learning method. Radar information gets coupled with navigational information for improved localization of objects during self-learning.

Keywords: Self-learning, Attention selection, Image categorization, Competition based learning, Mislabeling.

1 Introduction

Object recognition and categorization systems are integral components of the enterprise information system. We propose and illustrate an approach to augment supervised learning with self-learning in an object categorization system which uses images of various objects obtained with the help of sensor fusion.

Our self-learning approach may be useful in a variety of situations. For example, there may be situations in which object data arrives with missing labels but the constant presence of a skilled supervisor or an operator is simply impossible. The self-learning system can not only automatically categorize the objects but also continue learning about the object features even when the supervisor is not available.

Our specific focus is on automatic acquiring images from a mobile platform in an outdoor environment. We recognize the trend in automotive vehicles to combine sensors dedicated to various driver support, semi- and fully-autonomous functions. Active sensors such as radar/lidar systems demonstrate good performance of object detection in various outdoor environments. They may have difficulties with accurate object categorization due to insufficient resolution for extraction of object features. Passive sensors such as video cameras can provide sufficient resolution but they need to be guided for best performance. Considering complementary properties of these sensors, we can combine them in a single system for improved performance.

The fusion of data from radar and vision has been widely discussed for driver-assistance tasks (see, e.g., [1], [2], [3], [4], [5]). This work extends the previously published work [6] which detailed the radar-camera sensor fusion system coupled with the MILN algorithm for two-class object recognition. (The Multilayer In-place Learning

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 265–274, 2009.
© Springer-Verlag Berlin Heidelberg 2009

Network (MILN) is proposed by J. Weng et al. [7].) We also refer the reader to the companion publication [8] for the previous description of the self-learning system.

Published systems for object detection and recognition described elsewhere do not incorporate self-learning mechanisms. They operate generally in the following way. The system is calibrated or trained on the basis of many images of different classes of objects possible to encounter on the road during the system operation on board a vehicle. The calibration or training process is done prior to system deployment, and the system does not learn from its experience during deployment.

We propose a self-learning algorithm which enables a generic learning system to learn during its operation *without* intervention from the teacher/supervisor.

Our specific goal is to recognize objects in the path of a properly equipped vehicle. The system should reliably separate objects into two classes: "vehicles" and "non-vehicles", but it is not limited to two-class problems and can be easily extended to multi-class settings.

The detection and recognition/classification in our system is done on the basis of images segmented with the help of a radar. The radar guides the camera to extract what we termed attention windows in [6].

Given a 3D radar-returned target point projected on the image plane, attention windows are created within the original image, taking into account the expected maximum height and width of the vehicles. Such windows are of much smaller size than the original image, and they are centered around the radar return points. Figure 1 shows a few examples (strongest reflection is shown as the red dot inside each attention window shown as the blue rectangle).

The operation architecture of our system is shown in Figure 2. Two kinds of external (outward looking) sensors are employed: video cameras and radars. The radar provides the candidate attention points for the camera to zoom-in for better resolution. The

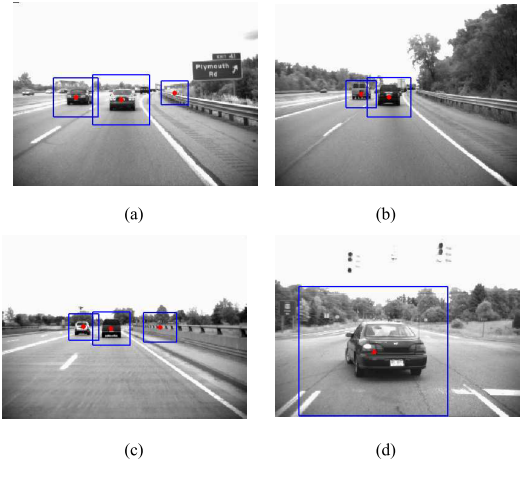


Fig. 1. Examples of images containing radar returns (red dots) which are used to generate attention windows (blue rectangles). This figure shows examples of different road environments in our experimental dataset.

attention points are shown as two red dots on the original image in the figure. In the case of the single camera the original image is taken first with the minimum zoom factor, followed by zooming-in on the first attention point (shown as the red dot pointed by the solid red arrow). An appropriate zoom factor can be applied readily based on the range measurement from the radar and the expected size of the target object. (No precise knowledge of the target object dimensions is necessary, as we intend only to capture a sufficiently detailed view of the object to make a reliable categorization (some "over-zooming" is acceptable).) The captured image is processed by a learning system for its categorization. The camera can then zoom in on the second attention point (shown as the red dot pointed by the dashed red arrow). The process is repeated for all attention points in the original image.

Multiple radars or cameras might be useful if a single radar and camera can not satisfy performance requirements (e.g., viewing angle, camera attitude speed). For example, one camera might only provide the original image, while the second camera functions in the zooming mode only. Though it is not illustrated in the figure, our self-learning mechanism also utilizes navigational information such as object locations on the map.

We just described the problem and the architecture of our system. In Section 2, we discuss the learning algorithm including both supervised and self-learning processes. We overview our experimental results in Section 3.

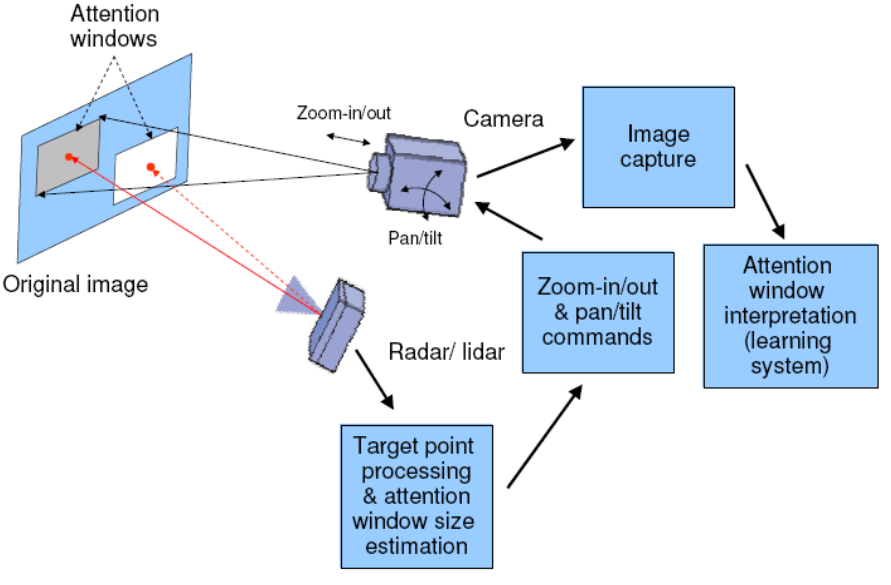


Fig. 2. Architecture of our system

2 Learning

The main idea of our self-learning is that the availability of reliable navigational information including map, global positioning system (GPS) and inertial measuring unit

(IMU) enables automatic selection of high-confidence class labels for subsequent training of an object recognition algorithm.

It can be inferred from geometric considerations and validated empirically that the object which returns a sufficiently strong radar/lidar reflection in a reasonably wide range can be reliably positioned on the map even if the road curves, provided that the ego-vehicles pose on a sufficiently accurate map is known. (A vehicle instrumented with the abovementioned sensors and the learning system is called ego-vehicle in Figure 3.)

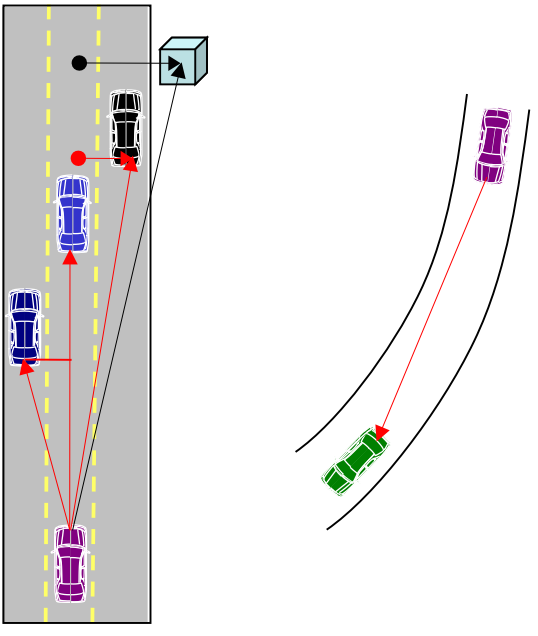


Fig. 3. Examples of common driving situations in which reasonably accurate localization of different objects in the vicinity of the ego-vehicle (shown as the source of all arrows, red for vehicles and black for a non-vehicle) is possible if the ego-vehicle itself is localized on a sufficiently accurate map

Our learning system can in principle be any generic learning algorithm which uses images of objects as its inputs. The learning system may acquire as its input one of the attention windows shown in Figure 1. The advantage of the (optical) zoom-in, as opposed to the simple extraction of the relevant part from the bigger original image implemented in [6], is the same level of image resolution for a distant objects as that for a nearby object. The same level of image resolution is of course advantageous for performance consistency (independence from distance to the object of interest). The main disadvantage is the need to have a special and potentially complex/expensive machinery for zooming in on the (possibly multiple) objects of interest registered on the original large image.

From the standpoint of learning it is convenient to use a system which operates on the principle of competition among its components (see Section 2.1 for our implemented

learning system). For example, the system may use the input image in a transformed space, i.e., not directly as pixel intensities but as some other features. For operational transparency, it is important that the transformed space be linked in one-to-one correspondence to the image space, i.e., the given image would correspond to a specific counterpart in the transformed space, and vice versa.

The generic learning system operates as follows. The system is initialized or trained with a collection of images with known and correct labels. When a new image is applied to the system's input, it is compared with stored images in the transformed space. For better comparison, it might be necessary to resort to a semantic analysis of the image, e.g., employing techniques elaborated in [10].

Each component of the system stores a unique template with its associated label. The template may be a composite of images in the transformed space for which the component turned out to be the winner of the competition among the components of the system. The component with the highest match with the input image wins, and its label is assigned to the new image. The winning component may adjust its template to better match its existing template with the new image. Each component also includes a group of counters. Each counter is assigned to a specific class of images, and it gets incremented whenever the component wins.

In addition to the object bearing and distance from the ego-vehicle, we also can take into account the relative speed of the object – another measurement typically provided by the radar. If the object is determined to be on the road and moving with speed comparable to that of the ego-vehicle, then with high confidence such object can be called a vehicle. If the object is determined to be off the road and moving in an opposite direction with respect to our vehicle, then such object can be provisionally labeled a non-vehicle. It may be convenient to utilize a special classifier or decision logic to help with the object labeling based on the object relative speed and location. For example, the fuzzy logic inference can be applied as illustrated in Figure 4. The object which has a large negative speed relative to the ego-vehicle speed V_e is likely a non-vehicle. Objects moving toward the ego-vehicle with even larger speeds ($\sim 2V_e$) are very likely vehicles.

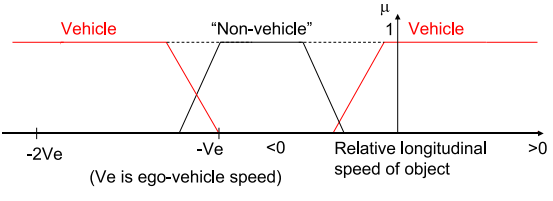


Fig. 4. A simple illustration of a decision logic based on linguistic variables (vehicle and provisional non-vehicle, or "non-vehicle"). The decision is based on computation of the degree of membership μ which depends on the relative longitudinal speed (negative/positive is moving toward/away from the ego-vehicle).

We can not guarantee that a stationary object identified to be off-road is always a non-vehicle, as it might be just a parked or broken vehicle. In order to avoid mislabeling such offroad objects, we utilize the abovementioned counters for each component storing the number of times the component wins in response to presentations of each class

of objects. If an unknown stationary object is detected by the radar, we compare the counters for the winning component and choose the class with the highest count.

To further decrease probability of mislabeling, the self-learning may start only on objects close-by, e.g., within 20 m from the ego-vehicle. In addition, other constraints on the object locations may need to be considered, e.g., objects which are located on approximately straight and flat regions of the road to avoid confusing reflections from overpasses.

2.1 LCA

The attention windows are fed into our classifier which is based on the Lobe Component Analysis (LCA) algorithm [9], which is a simple unsupervised form of the Multilayer In-place Learning Network (MILN) (single-layer unsupervised MILN). In [6] we compared two possible input representations for the MILN: pixels and sparse codes. (We implemented sparse coding by using Gabor-like orientation-selective filters.)

The learning system and its algorithm are shown in Fig. 5 and Algorithm 1.

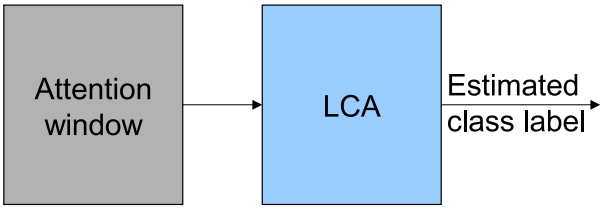


Fig. 5. The attention window (in the form of either pixel or sparse representation) is supplied as input to the LCA. The LCA provides the estimated class label.

Algorithm 1 is based on the LCA learning algorithm $\mathbf{z}(t) = \text{MILN}(\mathbf{s}(t))$, where $\mathbf{s}(t)$ is the input vector at time t (usually t corresponds to the presented attention window). The vector $\mathbf{z}(t)$ contains responses of neurons. We added counters $m_{j,l}$ to the base LCA algorithm to improve self-learning and automatic labeling of attention windows. Note that Algorithm 1 may learn continuously ($n \to \infty$ in step 2), if required.

The generic learning system, or the LCA in our current implementation, is intended to operate continuously. It is possible that the system may not be able to learn anything new because of memory overload.

The new memory can be added to the existing memory already allocated for the components of the learning system, assuming the memory itself is cheap and abundant. However, we still have to consider the consequences of the limited search speed in the growing memory. For example, if the categorization decision is required every 10 ms, then, for a 1 GFLOPS serial processor, only 500 components can be processed in time to decide the winner if each component requires 20,000 FLOPS for its processing.

The components in our generic learning system may need to be replaced or reinitialized in the course of the system's continued operation to relax memory or computational constraints. We illustrate the reinitialization process on our specific example of

Algorithm 1. Competition based neural network learning algorithm

1: Set: $c = 400$, age $n_j = 0$, counters $m_{j,l} = 0$ for all neurons j and classes l; $\mathbf{z} = \mathbf{0}$, the output of all neurons at time $t = 0$.
2: **for** $t = 1, 2, ..., n$ **do**
3: $\mathbf{y}(t) = \mathbf{s}(t)$;
4: **for** $i = 1, 2, ..., c$ **do**
5: Compute pre-response of neuron i from bottom-up connections:

$$\hat{z}_i = \frac{\mathbf{w_b}_i(t) \cdot \mathbf{y}(t)}{\|\mathbf{w_b}_i(t)\|\|\mathbf{y}(t)\|}$$

6: **end for**
7: Simulating lateral inhibition, decide the winner: $j = \arg\max_{1 \le i \le c}\{\hat{\mathbf{z}}_i(t)\}$.
8: Update the winner's counter: $m_{j,l} \leftarrow m_{j,l} + 1$ if the label or its estimate corresponds to the class l.
9: The 3×3 neighboring cells are also considered as winners and added to the winner set $\mathbf{J}$ for the subsequent updating of the weights $\mathbf{w_b}$.
10: The winner set $\mathbf{J}$ may still contain neurons with zero pre-responses $\hat{\mathbf{z}}$. Define a sub-set $\mathbf{J}' \subseteq \mathbf{J}$, such that the response $z_j = \hat{z}_j$ if $\hat{z}_j \neq \mathbf{0}, \forall j \in \mathbf{J}'$.
11: Compute $\mu(n_j)$ by amnesic function:

$$\mu(n_j) = \begin{cases} 0 & \text{if } n_j \le t_1, \\ c(n_j - t_1)/(t_2 - t_1) & \text{if } t_1 < n_j \le t_2, \\ c + (n_j - t_2)/r & \text{if } t_2 < n_j, \end{cases}$$

where $t_1 = 20, t_2 = 200, c = 2, r = 2000$ are example values of plasticity parameters.
12: Update neuron age n_j ($j \in \mathbf{J}'$): $n_j \leftarrow n_j + 1$ and weights of winners $j \in \mathbf{J}'$:

$$\mathbf{w_b}_j(t) = w_1 \mathbf{w_b}_j(t-1) + w_2 z_j \mathbf{y}(t)$$

where scheduled plasticity is determined by its two age-dependent weights w_1,w_2:

$$w_1 = (n_j - 1 - \mu(n_j))/n_j, w_2 = (1 + \mu(n_j))/n_j$$

13: For all $1 \le i \le c, i \notin \mathbf{J}'$, $\mathbf{w_b}_i(t) = \mathbf{w_b}_i(t-1)$.
14: **end for**

the learning system. The neurons in the LCA algorithm usually loose their ability for a major adaptation because of the dynamics of the weights w_1 and w_2. Indeed, the neuron becomes very conservative in responding to new data because w_2 approaches $1/r \ll 1$ while w_1 approaches $1 - 1/r \approx 1$ over time.

We can readily introduce the neuron regeneration in the LCA base algorithm. We can periodically reinitialize the neuron with the youngest age over a sufficiently large time window, in analogy with forgetting in biological memory. The reinitialized neuron can have its weights $\mathbf{w_b}$ (or its template) set to the current input image, and its parameters – to $n_j = m_{j,l} = 0$, $w_1 = 0$ and $w_2 = 1$.

Concluding this section, we summarize our system operation below:

• Carry out supervised (off-line or pre-deployment) learning for the LCA. Though the LCA is an unsupervised system, the individual counters $m_{j,l}$ need to be

associated with each class of the objects. The data set prepared for the supervised training is carefully inspected to avoid possible label confusions (when two different classes of objects are captured in the single attention window). Such carefully prepared data sets contain no labeling errors.

• Carry out self-learning:

A) Detect possible targets and verify which ones are on the road using maps and GPS/IMU.

B) Confirm if candidate vehicle targets are moving (use relative speed information).

C) Decide the type of stationary targets (compare the winning neuron's counters $m_{j,l}$ for all classes). (The decision on the type of the target based on the relative speed and positional information could also be confirmed using the counter comparison.)

D) Perform computations of the t loop in Algorithm 1 to decide the object class by comparing counters of the winning neuron. For the two-class problem, $m_{j,1} > m_{j,2}$ means that the object belongs to the first class, otherwise it belongs to the second class.

3 Experiments and Results

We used a properly equipped test vehicle to capture real-world image and radar/lidar sequences for training and testing. In [6] we compared our performance with the state of the art methods including the incremental SVM. Our system's performance was found to be very competitive.

The overall performance of the system is strongly dependent on the accuracy of camera guidance. It is clear that, if the camera is properly zoomed on the correctly labeled object, the recognition accuracy of the learning system will be higher for the system with the zoomed camera than for the system without any zooming. However, if the camera is zoomed incorrectly, for example, only an insufficiently small part of the object is visible in the attention window, then the performance may decrease. Furthermore, the label might not be correct either due to an incorrect placement of the object on the map or due to a mistake of the learning system (the neuron with the wrong label won the competition to determine the class of the object in the self-learning mode).

Methods that utilize both labeled and unlabeled data for training are known as semi-supervised learning. Due to specifics of our system, we opted for the following approach. We turned verification of self-learning into sensitivity verification, i.e., verification of how (in)sensitive our learning system is to mislabeling of objects (errors in labels) during a long training process.

We simulated self-learning by introducing labeling errors in otherwise the same supervised learning process of Algorithm 1. Our results indicate that performance in the presence of mislabeling errors is remarkably robust and exhibits only marginal degradation, as shown in Table 1. Specifically, each attention window in our experiment has a specified probability of its corresponding label to be changed to that of the opposite class.

It is reasonable to ask which level of mislabeling is acceptable. Based on a preliminary assessment, we estimate that the level of mislabeling will not exceed 30%. We train with different levels of mislabeling on up to 800 images employing random reshuffling of the sequence of images for up to 25,000 training steps or presentations of

different attention windows. We then test on 50 images not used in training, repeating the experiment 100 times for different random seeds. Repeated training on the same set of images is normal because it is inexpensive to retain many attention windows in the on-board memory.

The results in Table 1 are obtained for Algorithm 1 with the pixel input representation and the fixed learning rate ($w_1 = 1$, $w_2 = 0.003$) because the neurons are initialized with attention windows as templates. Such initialization assures good performance of the system before the self-learning begins.

We can observe no statistically significant differences between the results for zero and 10% mislabeling ($p \gg 0.05$ in the ANOVA test) for each respective number of training steps, as shown in the Table 1. The results also suggest that, while early performance of the algorithm with mislabeling degrades, under persistent mislabeling lasting substantial periods of time the performance actually improves, especially for 30% mislabeling.

We also carried our experiments when the neuron with the youngest age is reinitialized with the current image every 500 time steps. While the results with neuron reinitialization are little different from the results of Table 1 (e.g., the error rates are 2.5% and 0.54% in the case of 12,000 steps for 30% mislabeling and zero mislabeling, respectively), the question remains whether neurons responsible for recognition of rare objects may accidentally be "weeded out" in the process of reinitialization. As mentioned before, there is a tradeoff between the ability to learn new objects and the memory capacity, and it may be inevitable that some form of reinitialization has to be implemented in practice.

Table 1. Sensitivity to mislabeling (statistics over 100 experiments as average total error rate/ std/ max, %)

# train.steps	6000	12000	25000
Mislab.level, %	Stats, %	Stats	Stats
0	0.98/1.5/4	0.58/1.3/6	0.58/1.1/6
10%	0.76/1.3/6	0.58/1.3/6	0.54/1.1/6
20%	1.56/2.0/10	0.68/1.3/8	0.60/1.1/6
30%	6.1/3.9/18	2.3/2.2/10	0.92/1.3/6

Concluding this section, we make the following observation regarding apparent performance improvement with longer training in the presence of mislabeling. The performance illustrated in Table 1 would degrade substantially with fewer neurons in the network (parameter c). No visible improvement of performance would be observed with longer training because the smaller network quickly reaches its performance limits. For example, for $c = 100$ the average error rate is between 4.8% and 7%, increasing to 15% for $c = 25$. With more neurons in the network, each of them gets a chance to win the competition less often than in the case of smaller c. For higher levels of mislabeling, it takes longer for the network to self-correct the counters $m_{j,l}$ in Algorithm 1, i.e., to accumulate enough differences between $m_{j,1}$ and $m_{j,2}$ so that the correct class dominates over the incorrect class.

Relying on the counters to make the decision in the case of mislabeling is appealing because it features both robustness and performance improvement over time. Moreover, the training in Algorithm 1 remains unsupervised and independent of mislabeling, which is also important for robust self-learning.

4 Conclusions

We discussed an object categorization system based on the sensor fusion framework in which the radar/lidar directs attention of the camera and simplifies analysis required for each image.

The use of scanning/multi-beam instantaneous lidar as opposed to radar presents advantages such as availability of many more distance measurement points even for objects located relatively far from the ego-vehicle (~ 40m). Objects like passenger cars or SUVs usually have tens and hundreds of lidar measurements associated with them (even in a subset of heights) which helps to direct attention of the camera much more accurately than with the radar. Furthermore, the zooming cameras guided by the lidar can be helpful for recognition of traffic signs and more definitive recognition of pedestrians.

Properly localized targets enable self-learning in the system which is useful after its in-vehicle deployment. We verified the proposed self-learning mechanism in simulations with mislabeling and demonstrated its significant performance robustness on real-life data.

References

1. Coue, C., Fraichard, T., Bessiere, P., Mazer, E.: Multi-sensor data fusion using bayesian programming: An automotive application. In: International Conference on Intelligent Robots and Systems, Lausanne, Switzerland (2002)
2. Jochem, T., Langer, D.: Fusing radar and vision for detecting, classifying and avoiding roadway obstacles. In: Proceedings IEEE Symposium on Intelligent Vehicles, Tokyo (1996)
3. Grover, R., Brooker, G., Durrant-Whyte, H.F.: A low level fusion of millimeter wave radar and night-vision imaging for enhanced characterization of a cluttered environment. In: Proceedings 2001 Australian Conference on Robotics and Automation, Sydney (2001)
4. Laneurit, J., Blanc, C., Chapuis, R., Trassoudaine, L.: Multisensorial data fusion for global vehicle and obstacles absolute positioning. In: Proceedings of IEEE Intelligent Vehicles Symposium, Columbus (2003)
5. Miyahara, S., et al.: Target tracking by a single camera based on range-window algorithm and pattern matching. In: SAE 2006 World Congress and Exhibition, Detroit (2006)
6. Ji, Z., Prokhorov, D.: Radar-Camera Fusion for Object Classification. In: Proc. Fusion, Germany (2008)
7. Luwang, T., Weng, J., Lu, H., Xue, X.: A multilayer in-place learning network for development of general invariances. International Journal of Humanoid Robotics 4(2) (2007)
8. Prokhorov, D.: A Self-Learning Sensor Fusion System for Object Classification. In: Proc. IEEE Symposium Series on Computational Intelligence (SSCI), Workshop on CI in Vehicle and Vehicular Systems, Nashville, TN, USA, March 30-April 2 (2009)
9. Weng, J., Zhang, N.: Optimal in-place learning and the lobe component analysis. In: Proc. World Congress on Computational Intelligence, Vancouver, Canada, July 16-21 (2006)
10. Fei-Fei, L., Fergus, R., Perona, P.: Learning generative visual models from few training examples. Comput. Vis. Image Underst. 106, 59–70 (2007)

A Self-tuning of Membership Functions for Medical Diagnosis

Nuanwan Soonthornphisaj and Pattarawadee Teawtechadecha

Department of Computer Science, Faculty of Science, Kasetsart University
Bangkok, Thailand
fscinws@ku.ac.th, iq_phat@hotmail.com

Abstract. In this paper, a self-tuning of membership functions for fuzzy logic is proposed for medical diagnosis. Our algorithm uses decision tree as a tool to generate three kinds of membership functions which are triangular, bell shape and Gaussian curve. The system can automatically select the best form of membership function for the classification process that can provide the best classification result. The advantage of our system is that it doesn't need the expert to create membership functions for each feature. But the system can create various membership functions using learning algorithm that learns from the training set. In some domains, user can provide prior knowledge that can be used to enhance the performance of the classifier. However, in medical domain, we found that some diseases are difficult to diagnose. It would not be a problem if that disease has been completely explored in medical area. In order to rule out the patient, we need a domain expert to provide the membership functions for many attributes obtained from the laboratory test. Since the disease has not been completely explored in medical area, the membership function provided by the expert might be biased and lead to the poor classification performance. The performance of our proposed algorithm has been investigated on 2 medical data sets. The experimental results show that our approach can effectively enhance the classification performance compare to neural networks and the traditional fuzzy logic.

Keywords: Fuzzy logic, Gaussian curve, Triangular membership function, Bell Shape membership function, Medical diagnosis.

1 Introduction

In medical informatics field, concepts such as symptoms, signs, interpreted results obtained from the laboratory or from clinical investigations, usually identified by a linguistic term and can be formalized by fuzzy sets. In order to directly model these linguistic expressions, fuzzy sets allow us to assign degrees of compatibility of what is observed in the patient to what the term stands for may be applied.

Fuzzy logic is mainly contributed in two areas of medical fields which are medical control systems and expert systems. Fuzzy logic is very well suited for

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 275–286, 2009.
© Springer-Verlag Berlin Heidelberg 2009

medical control systems because the parameters involved are mostly uncertain. Hence a domain expert must provide the concept of symptoms that are obtained from the laboratory test in the form of ranged values. Many researches apply fuzzy logic as a key technology to deal with uncertainty and found that it provides a promising result. By the way, there are some diseases that haven't had enough knowledge about the symptom, yet. Therefore, using the traditional way of membership functions created by physician seems not to be a good idea. Because these membership functions created by human might be biased and lead to the wrong diagnosis.

Our hypotheses is that for the disease that is difficult to diagnose, we should do some kinds of data mining using machine learning technique to learn from set of attributes and used the knowledge to generate the membership functions automatically. The advantage of our idea is that the learning process can bring new knowledge from data that physicians may never know before and this will contribute to the body of knowledge for domain expert as well.

The rest of this paper is organized as follows: in Section 2, we briefly present the basic concepts of fuzzy logic and its contribution in many research papers. Section 3 provides the background of decision tree learning. In Section 4 we introduce the framework of our system. Section 5 shows all experimental results. We finally conclude the paper with an outlook to our future work in Section 6.

2 Basic Concept of Fuzzy Logic

In this section, we briefly review basic concepts of fuzzy sets which were introduced by Zadeh [17]. Fuzzy logic is the theory of fuzzy sets that calibrate vagueness. It is based on the idea that some attributes are obtained in degrees such as temperature, blood pressure, etc. These attributes are in the form of continuous values. Therefore, each attribute need a function called *membership function* to convert the continuous value to the degree of nominal value (ie. high, low). Fuzzy inference can be defined as a process of mapping from a given input to an output, using the theory of fuzzy sets. The most commonly used fuzzy inference technique is the so-called Mamdani method [8]. The Mamdani-style fuzzy inference process is performed in four steps which are fuzzification of the input variables, rule evaluation, aggregation of the rule outputs and finally the defuzzification process. The process of fuzzy inference consists of 4 steps which are as follows

Step1: Fuzzify inputs:

Step2: Apply Fuzzy Operator:

Step3: Apply Implication Method:

Step4: Aggregate All Outputs

Fuzzy logic has broadly been used in many medical informatics researches. An internal blood glucose monitoring system was developed by [13]. The system is able

to monitor the glucose levels and adjusts the level of insulin using an expert fuzzy logic algorithm. A specific software and hardware design can provide a high level of reliability. Their implementation uses multiple sensors to monitor blood glucose levels. The data will be compared against each other to control the amount of insulin to supply.

The severity of patients suffered from asthma, using a fuzzy decision-making analysis was determined by [7]. The data set is consists of two parts which are the objective severity (OS) (the standard tool for doctor) and the data obtained from questionnaire (PS). Both OS (rated by doctors) and PS (rated by patients) were rated as mild intermittent, mild persistent, moderate, or severe. These variables were pooled, and considered as potential variables patients might use to determine their PS. They were tested against the PS measurement using FDMA.

Four applications of fuzzy logic theory in epidemic problems were presented by [9], using linguistic fuzzy models, possibility measure, probability of fuzzy events and fuzzy decision making techniques. The results demonstrate that the application of fuzzy sets in epidemiology is a very promising area of research. Furthermore, fuzzy logic is used to enhance mammographic features for breast cancer diagnosis [3]. N.H. Phuong and V. Kreinovich build a fuzzy expert system for syndromes differentiation in oriental traditional medicine and combine with the disease diagnosis of Western Medicine [11]. Lesmo.L. et al. combines fuzzy production rules with frame-like structures in order to assess the liver function and to the diagnosis of hepatic disease [6].

Many hybrid systems combine fuzzy logic with neural networks such as the research done by [1]. They implemented a hybrid neuro-fuzzy prognosis system for the prediction of breast cancer relapse patients. The membership functions used in their works are the tumor sizes, grade of tumor and number of axillary nodes. Neuro-fuzzy was also implemented as a multi-sensor fusion system for control of dept of desflurane anesthesia. In this study, depth of desflurane anesthesia was examined through cardiovascular-based an adaptive neuro-fuzzy system according to the changing in the blood pressure and heart rate taken from the patient. The membership functions are heart rate and blood pressure which are provided by the physician [16].

An application that can induce atracurium for neuromuscular block during surgery was developed by [10]. They observed improved control over complex numerical techniques. The self-learning fuzzy control technique shows much promise for other medical applications such as post-operative blood pressure management, intra-operative control of anaesthetic depth, and multivariable circulatory management of intensive care patients.

Fuzzy logic was also used in Stuttering Therapy in order to correct the pronunciation of speech. A system called *Orator* can adjust all the therapy parameters automatically and tune them adaptively to the patient [5].

3 Decision Tree Learning

Decision tree learning [12] is a popular algorithm for supervised learning. The tree is constructed using only those best attributes that are able to differentiate the concepts of

the target class. Each node in the tree is an attribute selected from the training set using *gain ratio* (see Equation 2). The gain ratio measures the different between the *entropy* (see Equation 1) of the training set before and after selecting the attribute. The attribute which has the highest value of the gain ratio is selected to be a node in the tree.

$$\text{Entropy(S)} = \frac{-P}{P+N}\log_2\frac{P}{P+N} - \frac{N}{P+N}\log_2\frac{N}{P+N} \tag{1}$$

$$\text{Gain(S,A)} = \text{Entropy(S)} - \sum_{v \in \text{Values(A)}} \frac{|S_v|}{|S|}\text{Entropy}(S_v) \tag{2}$$

S = training set,
A = attribute,
S_v = subset of training data that has value v on attribute A,
P = training set with Positive class,
N = training set with Negative class.

4 The Self-tuning Algorithm

We extend our previous work [14] to make our algorithm more flexible in order to get higher performance by using variety styles of membership functions. The classification's performance is optimized by the most suitable form of membership functions. An approach called Self-tuning, which consists of the following steps:

1) The AutoGenFuzzy algorithm generates three kinds of membership function which are triangular, bell shape and Gaussian curve.
2) The membership tuning process that aims to locate the best degree of membership function belong to the intersection point (*m*).
3) The fuzzy inference is the final step that classifies the patients into classes.

4.1 Feature Selection

The feature selection algorithm is based on decision tree learning. Since the decision tree's mechanism aims to select the best attribute that has the highest information gain. Therefore, we develop a greedy feature selection algorithm as shown in Table 1.

We found that the selected features obtained from blood test are Intake, ALT, MCH, HCT, BUN, Mononuclearcells, CO2, AST and MCHC. The selected features obtained from symptoms are General Fatigue, Role Physical, Social Function, Bodily Pain, General Health and Vitality. Note that, the value found in all leave nodes are the class value.

The feature extraction obtained from WBCD data set are Single Epithelial Cell Size, Marginal Adhesion, Sample Code Number, Bland Chromatin and Uniformity of Cell Shape, respectively.

Table 1. Feature Selection Algorithm

```
Algorithm: Feature Selection
Initialize i = 1,  tmp = 0
FeatureSet = {a1, a2,…, an}
n = sizeofAttr(FeatureSet)
While i ≤ n
  acc = accuracy(DecisionTree(TrainData, FeatureSet))
 IF tmp ≤ acc  THEN
   tmp = acc
     FeatureSet = FeatureSet - {ai}
 IF tmp ≥ accuracy(DecisionTree(TrainData,FeatureSet))
THEN
     FeatureSet = FeatureSet + {a1}
  i++
END WHILE
RETURN FeatureSet, DecisionTree
```

4.2 Automatic Membership Functions Generation

The AutoGenFuzzy algorithm generates a set of membership functions for each attribute using the decision tree. In this study, three kinds of membership functions are automatically generated which are triangular, bell and Gaussian curve.

Considering any $attribute_i$ founded in the decision tree, the AutoGenFuzzy creates the number of membership functions (graph) of that attribute using Equation 3. That means the number of membership functions for $attribute_i$ depends on the number of occurrences of this attribute appears in the tree plus one.

$$G(attribute_i) = \text{NumberofNode}(attribute_i) + 1 \tag{3}$$

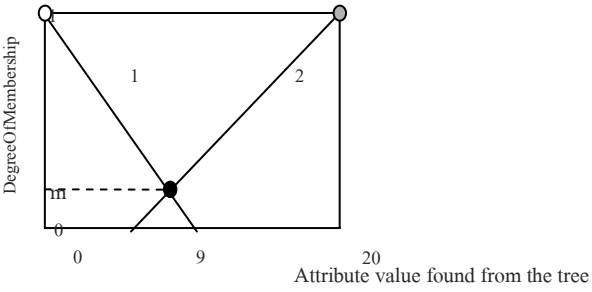
where as G($attribute_i$) is the number of graphs

NumberofNode ($attribute_i$) is the number of occurrences of $attribute_i$.

Next, the membership function of that attribute is generated using the lowest value of that attribute founded in the decision tree and we assign the degree of membership as 1. For the highest value of this attribute, the AutoGenFuzzy algorithm generates another graph and assigns the degree of membership as 1.

4.2.1 The Attribute Occurs Only Once in the Decision Tree

For any $attribute_i$ occurs only once in the tree, there are 2 graphs for such attribute (see Figure 1 and 2). Suppose that the graph's line is connected in triangular form (see Figure 1)



Where as
= The lowest value of attribute$_i$
= The highest value of attribute$_i$
= The value of attribute$_i$ found in the decision tree

Fig. 1. The membership functions for *attribute*$_i$ which has one occurrence in the decision tree

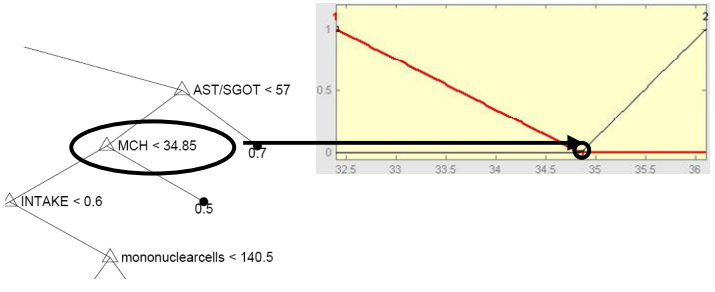


Fig. 2. The membership function of attribute *MCH*

4.2.2 The Attribute Occurs Twice in the Decision Tree

In case that the attribute founds in the decision tree twice. The algorithm generates three membership functions (see Figure 3-4). The AutoGenFuzzy algorithm is shown

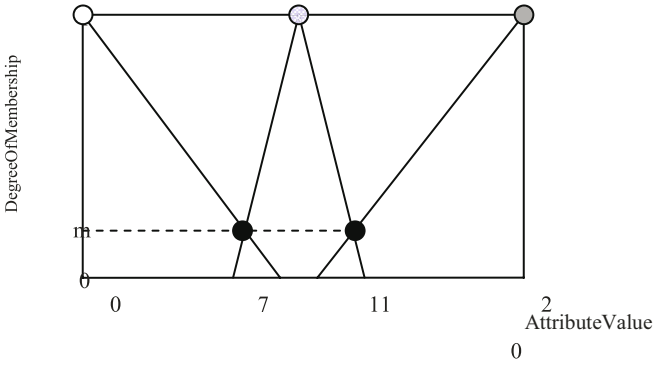


Fig. 3. The membership function of attribute which has 2 occurrences founded in the decision tree

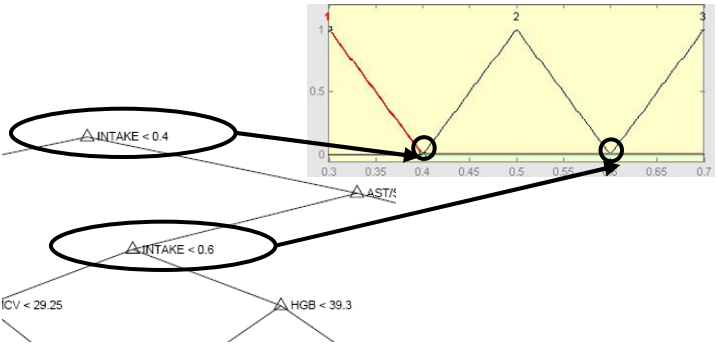


Fig. 4. The membership function of attribute INTAKE which has 2 occurrences founded in the decision tree

Table 2. Self-tuning algorithm

```
Algorithm: Self-tuning

INPUT: TrainingSet,FeatureSet
MemFn ={Triangular,Bell-shape Gaussian }
BEGIN
FOR i=1 TO 3
 Performance[i]=   Classify(AutoGenFuzzy(Memfn[i]))
END FOR
IF MAX(Performance)== Performance[1]) THEN
   MF = MemFn[1]
IF MAX(Performance) == Performance[2])THEN
   MF = MemFn[2]
IF MAX(Performance) == Performance[3])THEN
   MF = MemFn[3]

END
```

in Table 3. The algorithm scan the decision tree in order to get a set of features and count the number of their occurrences in order to determine the number of membership functions for each attribute.

4.3 Membership Function Tuning

The intersection point tuning is an optional step since the AutoGenFuzzy has already set the default of *m* to be 0.2 which provide the promising results. This step makes fine tuning for membership function. The process aims to locate the intersection point (*m*) between each graph line of membership function. The range of value *m* is between 0-0.9. We evaluate the affect of different values of *m* by doing fuzzy classification.

Table 3. Membership function generation algorithm

Algorithm: AutoGenFuzzy

```
Input: DecisionTree
Atr = getTreeNodes(DecisionTree)
m = 0.2
FOR i=1 to TreeSize

  Count = NoOfoccurence(attribute_i)
  IF  Count > 1 THEN
  FOR j=1 to Count+1
    IF  j = 1 THEN
      b = min(Atr_i.value )
      c=locateIntersect(b, Atr_i.value[j] ,m)
      a = b-c
      DrawGraph(a,b,c)
     ELSE IF  j = Count+1
          b = max(Atr_i value)
          c = locateIntersect(b,Atr_i.value[j-1] ,m)
          c = b+a
          DrawGraph(a,b,c)
   ELSE
          b = (Atr_i.value[j-1] + Atr_i.value[j]) / 2
          a = locateIntersect(b, Atr_i.value[j] ,m)
          c = locateIntersect(b, Atr_i.value[j] ,m)
          DrawGraph(a,b,c)
   END FOR
 ELSE
 b = min(Atr_i)
 c = locateIntersect(b, (min(Atr_i.value)+max(Atr_i.value))/2 ,m)
 a = b-c
 DrawGraph(a,b,c)
 b = max(Atr_i)
 a=locateIntersect(b, (min(Atr_i.value)+max(Atr_i.value))/2 ,m)
 c = b+a
 DrawGraph(a,b,c)
 END IF
 END FOR
RETURN MembershipFunctions
```

5 Experiment

5.1 Data Sets

Two data sets were used in our experiments which are the Chronic Fatigue Syndrome and the Wisconsin Breast Cancer data set.

5.1.1 The Chronic Fatigue Syndrome Data Set

This data set was obtained from CDC Chronic Fatigue Syndrome Research Group. There are 3 classes of patients as follows:

a) Patients who suffered from Chronic Fatigue Syndrome disease (54 cases). The class is labeled as CFS.
b) Patients who were ruled out as normal ones (64 cases). The class is labeled as NF.
c) Patients who have insufficient number of symptoms or fatigue severity (69 cases). The class is labeled as ISF.

We separate patients' records into two groups which are the blood test data and the symptom data obtained from patient's self-report. The blood test data contains 34 blood test values. The symptom data obtained from self-report consists of 70 attributes.

5.1.2 The Wisconsin Breast Cancer Data Set (WBCD)

The data set consists of 683 samples that were collected by Dr. W.H. Wolberg at the University of Wisconsin-Madison Hospitals taken from needle aspirates from human breast cancer tissue. The WBCD consists of nine features obtained from fine needle aspirates. 444 of the data set with 683 samples belong to benign, and the remaining 239 samples are of malignant class. The malignant means that the patient gets cancer pathology whereas the benign implies that the patient is healthful.

5.2 Experimental Results

A comprehensive performance study has been conducted to evaluate our method. In order to make the experimental results more reliable, we use *10*-fold cross validation method. All experimental results are evaluated using sensitivity, specificity and accuracy, respectively. In this section, we describe those experiments and the results. We run our algorithm on 2 datasets which are CFS and WBCD to test its classification performance against traditional Fuzzy logic and Neural Networks.

5.2.1 Performance of AutoGenFuzzy Using Different Feature Set

The objective of this experiment is to evaluate the potential of our feature selection algorithm. In order to test the feature selection algorithm, we set up the experiment as follows:

1) We used 2 data sets which are CFS and WBCD data set.

2) We run two different learning algorithms which are AutoGenFuzzy and Backpropation Neural Networks.

From Table 4, we can see that feature selection provides satisfactory results. Considering the CFS data set, we found that feature selection can enhance the performance of both Neural Networks and Self-tuning. However, Neural Network gets less performance compared to Self-tuning on both data sets.

Table 4. Performance obtained from AutoGenFuzzy on CFS and WBCD data set

Data Set	Algorithm	FeatureSelection	Accuracy
CFS	NeuralNetworks	yes	67.50
		no	46.22
	AutoGenFuzzy	yes	87.40
		no	83.37
WBCD	NeuralNetworks	yes	95.19
		no	95.19
	AutoGenFuzzy	yes	97.10
		no	95.47

5.2.2 Performance of Self-tuning Using Different Forms of Membership Function

We compare the performance of Self-tuning algorithm using different forms of membership functions which are triangular, bell shape and Gaussian curve. Classification performance on CFS and WBCD data set are shown in Table 5 and 6.

Table 5. Performance obtained from Self-tuning on CFS data set

Membership functions		Class Label		
		NF	ISF	CFS
Triangular	Sensitivity	100.00	78.57	70.33
	Specificity	100.00	86.43	88.79
	Accuracy	100.00	83.50	83.50
Bell shape		NF	ISF	CFS
	Sensitivity	93.57	67.14	75.67
	Specificity	96.73	85.39	85.77
	Accuracy	95.67	79.06	82.89
GaussianCurve		NF	ISF	CFS
	Sensitivity	100.00	70.00	72.67
	Specificity	97.57	87.48	86.65
	Accuracy	98.39	81.00	82.61

Considering the results shown in Table 6, we found that Self-tuning using triangular membership function can effectively rule out the patients who are not suffered from the disease (NF). Self-tuning gets 100% of sensitivity, specificity and accuracy.

For the CFS patients, we found that AutoGenFuzzy using bell shape membership function gets 75.67% on sensitivity which is more than that of the traditional fuzzy which gets only 59.67%. Considering the specificity performance, Self-tuning using triangular gets 88.79% which is lower than that of traditional fuzzy logic which gets 100%. However, this performance is acceptable because in medical domain the

Table 6. Performance obtained from Self-tuning on WBCD data set

Membership functions		Class Label	
		Malignant	Benign
Triangular	Sensitivity	91.80	94.73
	Specificity	94.73	91.80
	Accuracy	94.03	94.03
Bell shape		Malignant	Benign
	Sensitivity	92.34	95.93
	Specificity	95.93	92.35
	Accuracy	94.91	94.91
GaussianCurve		Malignant	Benign
	Sensitivity	91.54	94.77
	Specificity	94.77	91.54
	Accuracy	93.85	93.85

specificity means the ability to rule out the CFS patient which is less important than the ability to detect the real CFS patient measured by the sensitivity. Therefore, the overall performance of Self-tuning is more satisfiable than the traditional one since the Self-tuning is more sensitive than the traditional fuzzy logic.

Considering the sensitivity in Table 6, we found that the Self-tuning using Bell shape membership function outperforms other kinds of membership function (sensitivity = 92.34%). The ability of Self-tuning to rule out the malignant patient is 95.93%

6 Conclusions

In this study, we propose an approach that can automatically generate membership functions without domain expert intervention. The advantage of our system is that it can enhance the classification performance because of the contribution of the decision tree learning. The learning algorithm creates the concept of CFS data set by selecting informative feature sets and partitions each feature value that maximize the information gain. Therefore, we build an automatic system that gets the learned decision tree as input and generate the set of membership functions as output. We test the potential of these membership functions and compare to the traditional fuzzy logic using the well-known membership function obtained from domain expert. We compared our classification performance in terms of sensitivity, specificity and accuracy to the traditional fuzzy logic algorithm. We found that Self-tuning algorithm is a high potential algorithm since it outperforms the traditional fuzzy logic and Neural Networks. We plan to investigate the self-tuning algorithm on other data sets in the near future.

References

1. Aragonés, J.M.J., Sánchez, J.I.P., Doña, J.M., Alba, E.: A Neuro-fuzzy decision model for prognosis of breast cancer Relapse. In: Conejo, R., Urretavizcaya, M., Pérez-de-la-Cruz, J.-L. (eds.) CAEPIA/TTIA 2003. LNCS, vol. 3040, pp. 638–648. Springer, Heidelberg (2004)
2. CFS Data set, `http://www.cdc.gov/ncidod/diseases/cfs/`
3. Cheng, H.D., Xu, H.: A novel fuzzy logic approach to mammogram contrast enhancement. Information Sciences 148(1-4), 167–184 (2002)
4. Cox, E.: The Fuzzy systems Handbook: A practitioner's Guide to building, using and Maintaining Fuzzy Systems. Academic Press, Cambridge (1994)
5. Kwasnicka, H., Zak, B.: Fuzzy logic in stuttering therapy. In: Rutkowski, L., Tadeusiewicz, R., Zadeh, L.A., Żurada, J.M. (eds.) ICAISC 2006. LNCS, vol. 4029, pp. 925–930. Springer, Heidelberg (2006)
6. Lesmo, L., Saitta, L., Torasso, P.: Dealing with uncertain knowledge in medical decision-making: A case study in hepatology. Artificial Intelligence in Medicine 1(3), 105–116 (1989)
7. Lurie, A., Marsala, C., Hartley, S., Bouchon-Meunier, B., Dusser, D.: Patients perception of asthma severity, Respiratory Medicine, July 23 (2007)
8. Mamdani, E.H., Assilian, S.: An experiment in linguistic synthesis with a fuzzy logic controller. Int. J. of Man-Machine Studies 7(1), 1–13 (1975)
9. Massad, E., Regina, N., Ortega, S., Struchiner, C.J., Burattini, M.N.: Fuzzy epidemics. Artificial Intelligence in Medicine 29(3), 241–259 (2003)
10. Mason, D.G., Linkens, D.A., Edwards, N.D.: Self-learning fuzzy logic control in medicine. In: Keravnou, E.T., Baud, R.H., Garbay, C., Wyatt, J.C. (eds.) AIME 1997. LNCS, vol. 1211, pp. 300–304. Springer, Heidelberg (1997)
11. Phuong, N.H., Kreinovich, V.: Fuzzy logic and its applications in medicine. International Journal of Medical Informatics 62(2-3), 165–173 (2001)
12. Quinlan, J.R.: Introduction of Decision Trees. Machine Learning 1, 81–106 (1986)
13. Shereck, D., Jabur, F.: VHDL implementation of a fuzzy logic based expert system to control insulin-pump doses. McGill University, ECE Department, pp. 304–487 (2005)
14. Soonthornphisaj, N., Kijsirikul, B.: An Automatic Membership Functions Generation using Decision Tree for Fuzzy Logic. In: Proc. of the 8th Int. Conf. Intelligent Technologies. pp. 87–94 (2007)
15. WBCD Data set: Blake, C.L., Merz, C.J.: UJI Repository of Machine Learning Databases, `ftp://ftp.ics.uci.edu/pub/machine-learning-databases` (last accessed: June 15, 2006)
16. Yardimci, A., Hadimioglu, N., Bigat, Z., Ozen, S.: Depth Control of Desflurane Anesthesia with an Adaptive Neuro-Fuzzy System. Advances in Soft Computing 2, pp. 787–796. Springer, Heidelberg (2005)
17. Zadeh, L.A.: Fuzzy sets. Inf. Control 8, 338–353 (1965)

Insolvency Prediction of Irish Companies Using Backpropagation and Fuzzy ARTMAP Neural Networks

Anatoli Nachev[1], Seamus Hill[1], and Borislav Stoyanov[2]

[1] Business Information Systems, Cairnes School of Business & Economics
NUI, Galway, Ireland
{anatoli.nachev,seamus.hill}@nuigalway.ie
[2] Department of Computer Systems and Technologies, Shumen University, Bulgaria
bpstoyanov@abv.bg

Abstract. This study explores experimentally the potential of BPNNs and Fuzzy ARTMAP neural networks to predict insolvency of Irish firms. We used financial information for Irish companies for a period of six years, preprocessed properly in order to be used with neural networks. Prediction results show that with certain network parameters the Fuzzy ARTMAP model outperforms BPNN. It outperforms also self-organising feature maps as reported by other studies that use the same dataset. Accuracy of predictions was validated by ROC analysis, AUC metrics, and leave-one-out cross-validation.

Keywords: Insolvency prediction, Data mining, Neural networks, Backpropagation, Fuzzy ARTMAP.

1 Introduction

One of the most significant threats for many businesses today, despite their size and the nature of operations, is insolvency. Some evidences show that in the past two decades business failures have occurred at higher rates than at any time since the early 1930s [6]. Certain sectors of the economy, such as small industrial businesses in depressed areas, experienced failure rates as high as 50% over a five-year period [12]. The economic cost of business failures is significant as market values of the distressed firms declines substantially prior to their collapse [5]. Suppliers of capital, investors and creditors, as well as management and employees, are severely affected by business failures. The need for reliable empirical models that predict corporate insolvency promptly and accurately is imperative to enable the parties concerned to take either preventive or corrective action.

Kumar and Ravi [10] outline techniques for financial diagnosis, insolvency, and bankruptcy prediction, grouped into two broad categories - statistical and intelligent. The statistical techniques include: linear discriminant analysis; multivariate discriminate analysis; quadratic discriminant analysis; logistic regression (logit); and factor analysis. The group of intelligent techniques include different types of neural networks (NN), such as backpropagation NN (BPNN); probabilistic NN; auto-associative NN; self-organizing feature maps (SOFM); cascade correlation NN;

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 287–298, 2009.
© Springer-Verlag Berlin Heidelberg 2009

adaptive resonance theory NN [11]. Techniques also include; decision trees; case-based reasoning; evolutionary approaches; rough sets; soft computing (hybrid intelligent systems); operational research techniques including linear programming; data envelopment analysis; quadratic programming; support vector machines; fuzzy logic, etc .

In their study, Balcaen and Ooghe [3] found many difficulties in performance of the statistical techniques due to data anomalies, inappropriate sample selection, matters related to non-stationarity and instability of the data, inappropriate selection of independent variables and wrong consideration of the influence of time in the modelling.

Atiya [2] concluded in his research that, in general, the neural networks outperform statistical techniques and suggested to try to improve the predictive ability of the networks.

The objective of this study is to explore the potential of BPNNs and Fuzzy ARTMAP NN to provide insolvency warning signals for Irish firms. We use a recent Irish dataset for experiments and compare our results with those from another study [9] which used the same dataset, but different prediction techniques.

The paper is organized as follows: section 1 introduces the insolvency prediction problem and outlines previous research; BPNNs and Fuzzy ARTMAP NNs are presented in sections 2 and 3; section 4 discusses the dataset data pre-processing techniques; the empirical results are presented and analysed in section 5; conclusions are presented in section 6.

2 Backpropagation Neural Networks

A BPNN consists of a set of input nodes that constitute the input layer, one or more hidden layers of neurons, and an output layer of neurons. Figure 1 shows the BPNN architecture used in this study. The layers are fully interconnected. Connections between neurons have associated weights w_{ij} that express the relative importance of their input, and ultimately, form the output. The hidden and output layers have also a bias connection (squares in figure 1) with value 1 and weight θ.

Size of the input layer corresponds to the size of input patterns. As we use the BPNN for binary classifications, the output layer consists of one node only. Finding an optimal size of the hidden layer is a general problem with all BPNN. If the size is too small, the network will be unable to model complex data, and the resulting fit will be poor. If too many nodes are used, the training time may become excessively long, and worse, the network may overfit the data. There is no theoretical ground to suggest that size, but some heuristics are in use, such as the formula (1) we used.

$$n_h = \frac{n_s}{\alpha(n_i + n_o)} \tag{1}$$

where n_h is the size of the hidden layer; n_s is the number of training samples; n_i and n_o are the size of the input and output layers respectively; $\alpha \in [5, 10]$ is a scaling factor, smaller of noisy data and larger for relatively less noisy data. Our estimate of n_h is between 2 and 3, so we experimented with two BPNN: BPNN2 with 2 hidden nodes and BPNN3 with 3 nodes.

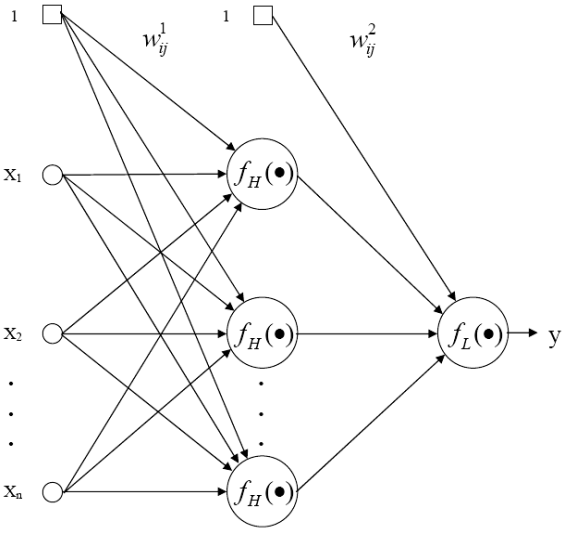


Fig. 1. Architecture of a backpropagation neural network with one hidden layer

Each hidden and output node computes its activation level by (2).

$$s_i = \sum_j w_{ij} x_j + \theta \tag{2}$$

The hidden nodes output using hyperbolic tangent sigmoid activation function.

$$f_H(x) = \frac{e^{2x} - 1}{e^{2x} + 1} \tag{3}$$

The output node uses log-sigmoid activation function (4).

$$f_L(x) = \frac{1}{1 + e^{-\beta x}} \tag{4}$$

where β is the slope parameter.

We trained the BPNN by Levenberg-Marquardt (LM) backpropagation (BP) algorithm. LM is a second-order nonlinear optimization technique that uses an approximation to the Hessian matrix. It was chosen from the various BP training algorithms as it trains a moderate size neural network 10 to 100 times faster than the usual gradient descent backpropagation method and produces better results [8, 14].

3 Fuzzy ARTMAP Neural Network Classifier

Fuzzy ARTMAP (FAM) architectures are neural networks designed to classify real-valued input patterns in real time [4]. A FAM network consists of two Fuzzy ART networks, ARTa and ARTb, bridged via an inter-ART module (see figure 2) and is

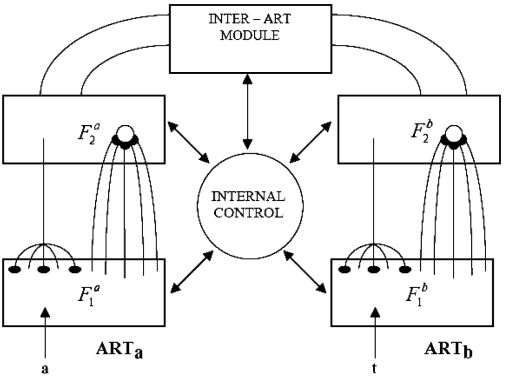


Fig. 2. Simplified Fuzzy ARTMAP architecture

capable of forming associative maps between clusters of its input and output domains in a supervised manner. The Fuzzy ART network consists of two fully connected layers of nodes: an M-node input layer F1 and an N-node competitive layer F2. A set of real-valued weights (black circles in the figure) is associated with bottom-up F1-to-F2 layer connections and top-down F2-to-F1 layer connections. Each F2 node represents a recognition category that learns a prototype vector. The F2 layer is connected, through learned associative links, to the inter-ART module.

FAM classifiers perform supervised learning of the mapping between training set vectors $\boldsymbol{a}$ and output labels $\boldsymbol{t}$, where $\boldsymbol{t_K}$ =1 if $\boldsymbol{K}$ is the target class label for $\boldsymbol{a}$, and zero elsewhere. The following algorithm describes FAM functioning [4], [7]:

1. *Initialization* of weights and network parameters. Initially, all F2 nodes are uncommitted.
2. *Input Pattern Coding*. When a training pair ($\boldsymbol{a, t}$) is presented to the network, $\boldsymbol{a}$ undergoes a transformation called complement coding, which doubles its size and results in a pattern $\boldsymbol{A}$. The vigilance parameter ρ is reset to its baseline value.
3. *Prototype Selection*. A pattern $\boldsymbol{A}$ activates layer F1 and propagates through weighted connections to layer F2. Activation of each node $\boldsymbol{j}$ in the F2 layer is determined by (5):

$$T_j(A) = \frac{|\, A \wedge w_j \,|}{\alpha + |\, w_j \,|} \tag{5}$$

where $\wedge$ is the fuzzy AND operator. The F2 layer produces a binary, winner-take-all (WTA) pattern of activity such that only the node $\boldsymbol{j=J}$ with the greatest activation value remains active. Node $\boldsymbol{J}$ propagates its top-down expectation back onto F1 and the vigilance test is performed. This test compares the degree of match between the expectation $\boldsymbol{w_J}$ and $\boldsymbol{A}$ against the dimensionless vigilance parameter $\boldsymbol{\rho}$ (6):

$$\frac{|\, A \wedge w_J \,|}{M} \geq \rho \tag{6}$$

If the test is passed, then node ***J*** remains active and resonance occurs. Otherwise, the network inhibits the active F2 node and searches for another node ***J*** that passes the vigilance test. If such a node does not exist, an uncommitted F2 node becomes active and undergoes learning.

4. *Class prediction*. Pattern ***t*** is fed to the inter-ART module, activated by the F2 category of ARTa as well. The module produces a binary pattern of activity in which the most active node is ***K***. If that node constitutes an incorrect class prediction, then a match tracking signal changes the vigilance parameter just enough to induce another search among F2 nodes in step 3. This search continues until either an uncommitted F2 node becomes active, or a node ***J*** that has previously learned the correct class prediction becomes active.
5. *Learning*. Learning input ***a*** updates the prototype vector $\boldsymbol{w_J}$ by (7):

$$w'_J = \beta(A \wedge w_J) + (1-\beta)w_J \tag{7}$$

where β is a fixed learning rate parameter.

4 The Data

The dataset contains financial information for a period of six years for a total of 88 Irish firms, of which 44 are insolvent and 44 are solvent. The dataset consists of Altman's [1] financial ratios as they have been the most widely and consistently used to date by both researchers and practitioners. The ratios are:

- R1: Working Capital / Total Assets;
- R2: Retained Earnings / Total Assets;
- R3: Earnings Before Interest and Taxes (EBIT) / Total Assets;
- R4: Market Value of Equity / Book Value of Total Debt;
- R5: Sales / Total Assets.

The working capital is current assets minus the current liabilities, which is an indication of the ability of the firm to pay its short term obligations. A firm's total assets are sum of the firm's total liabilities and shareholder equity. It can be viewed as an indicator of its size and therefore can be used as a normalizing factor. The retained earnings is the surplus of income compared to expenses, or total of accumulated profits since the firm commencement. The firm's earnings before interests and taxes is also an important indicator. Low or negative earnings indicate that the firm is losing its competitiveness, and that endanger its survival. Market capitalization relative to the total debt indicates that a firm is able to issue and sell new shares in order to meet its liabilities. Total sales of a firm, relative to the total assets, is an indicator of the health of its business, but without certainty as it can vary a lot from industry to industry.

Our research uses the Altman's ratios with two changes necessitated: operating profit was used instead of profit before interest and tax and so may contain a negligible amount of interest receivable; total shareholder funds was used as a proxy for market value of equity because not all of the companies used were quoted.

We used both a dataset with all Altman's ratios and one with a reduced number of variables. Reduction of variables has a potential to improve the NN abilities to

classify and alleviate the effect of the curse of dimensionality problem that appears with small datasets. This is because a NN with fewer inputs has fewer adaptive parameters to be determined, and these are more likely to be properly constrained by a data set of limited size, leading to a network with better generalization properties. In addition to that, a network with fewer weights may be faster to train. A variable selection also helps to avoid the overfitting phenomenon that makes NN to adjust to very specific random features of the training data that have no causal relation to the target function and makes the NN to lose its ability to generalize.

A F-ratio analysis shows that variables can be scored by their discriminatory power and that the set of variables [R1, R2, R3, R4] is a good selection. The same selection was used by Serrano [13] with an American dataset and Jones [9] with the dataset we use in this study. This also allows our results to be compared with those from the other studies.

5 Empirical Results and Discussion

We experimented with BPNN2, BPNN3, and Fuzzy ARTMAP NN simulators as predictors of insolvency of Irish companies. The NN were trained and tested in order to map testing instances to either positive or negative class labels {p, n}. Classification outcomes of each experiment were recorded in a confusion matrix that contains four types of predictions: true positive (TP), or positive hits; true negative (TN), or correct rejections; false positive (FP) or type I error; and false negative (FN), or type II error. In order to estimate the classifier performance we used the following metrics:

- accuracy (ACC): $ACC = (TP + TN) / (P + N)$ (8)
- true positive rate (TPR): $TPR = TP / (TP + FN) = TP / P$ (9)
- false positive rate (FPR): $FPR = FP / (FP + TN) = FP / N$ (10)

Most common estimate for a classifier is ACC. It represents the total number of correctly classified instances divided by the total number of all available instances. ACC, however, can be misleading in applications where important classes are underrepresented in datasets, (class distribution is skewed), or if errors of type I and type II can produce different consequences and have different cost. Secondly, the accuracy depends on the classifier's operating threshold, such as threshold values of BPNN or vigilance parameter of Fuzzy ARTMAP NN and choosing the optimal threshold can be challenging. Those ACC deficiencies can be addressed by the Receiver Operating Characteristics (ROC) analysis, discussed later in this section, which involves TPR and FPR.

5.1 Prediction Accuracy

BPNN2 and BPNN3 are soft classifiers as they use log-sigmoid activation function in the output node. A soft (probabilistic) classifier outputs estimates of the confidence of prediction. BPNN2 and BPNN3 output ranks between 0 and 1 instead of crisp class membership required by the task objectives. Conversion of ranks into crisp true/false

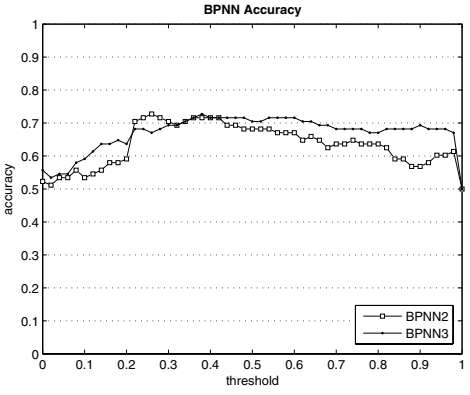


Fig. 3. Prediction accuracy of BPNN classifiers with 2 and 3 hidden nodes

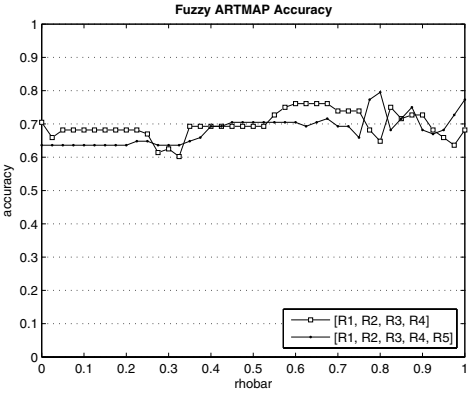


Fig. 4. Prediction accuracy of Fuzzy ARTMAP NN classifiers with [R1, R2, R3, R4] and [R1, R2, R3, R4, R5] sets of Altman's ratios varying vigilance parameter from 0 to 1 with increment of 0.025

values can be achieved by a threshold function with a certain threshold value. Varying the value, however, produces different outcomes, and ultimately different classifiers. Selection of an optimal classifier involves finding the relation between the threshold values and classification accuracy. Our experiments showed that best accuracy of BPNN2 can be achieved with threshold value 0.26 (72.7%); optimal threshold value for BPNN3 is 0.38 (72.7%). Experimental results are illustrated in figure 3. This makes us to conclude, that in terms of accuracy BPNN2 and BPNN3 perform equally well, but with different threshold functions. Apparently, the size of the hidden layer (2 or 3) is not essential for obtaining best accuracy.

Another series of experiments sought to estimate the prediction accuracy of the Fuzzy ARTMAP classifier. In order to investigate how vigilance parameter relates to the prediction accuracy, the NN was trained 41 times with vigilance parameter values from 0 to 1 with increment of 0.025. Results, illustrated in figure 4, show that the NN

achieves accuracy 76.1% with reduced variable dataset and vigilance parameter values between 0.6 and 0.675; full set of Altman's ratios and vigilance parameter value 0.8 provide accuracy 79.5%. Results show that Fuzzy ARTMAP NN can outperform both BPNN2 and BPNN3, and also SOFM (77.27%) used by Jones [9] with the same dataset. Outcomes also suggest that reduction of the Altman's variables worsen performance of the Fuzzy ARTMAP NN.

In accordance with certain deficiencies of the prediction accuracy discussed above, we did ROC analysis of the results.

5.2 ROC Analysis

ROC curves describe the relation between two indices: true positive rate (TPR) and false positive rate (FPR) as defined above. A ROC curve plots a point for every possible decision threshold imposed on the decision variable and depicts relative trade-offs between benefits (true positives) and costs (false positives). Figure 5 shows different BPNN2 and BPNN3 classifiers obtained by varying the threshold value from 0 to 1. The two ROC curves are step functions. By providing such a complete picture, ROC curves can be used to select the optimal decision threshold by maximizing any pre-selected measure of efficacy (e.g., accuracy, average benefit, etc.). In general, the best possible prediction method would yield a point in the upper left corner or coordinates (0, 1) of the ROC space, representing 100% sensitivity (all true positives are found) and 100% specificity (no false positives are found). A completely random guess would give a point along a diagonal line (line of no-discrimination) from the left bottom to the top right corners. Analysis of the results shows that the best classifier of BPNN3 is at point A; best of BPNN2 is at point B (see figure 5). This is because the two points are most 'northwest', or most distant from the no-discrimination line. Points A and B represent two classifiers with fixed threshold values 0.26 and 0.38 respectively, and both provide accuracy of 72.7%. These two

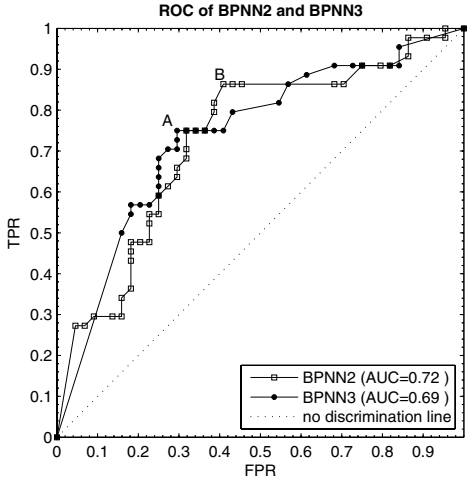


Fig. 5. ROC curves of BPNN classifiers with 2 and 3 hidden nodes. Points A and B represent maximal performance at threshold values 0.38 and 0.26 respectively.

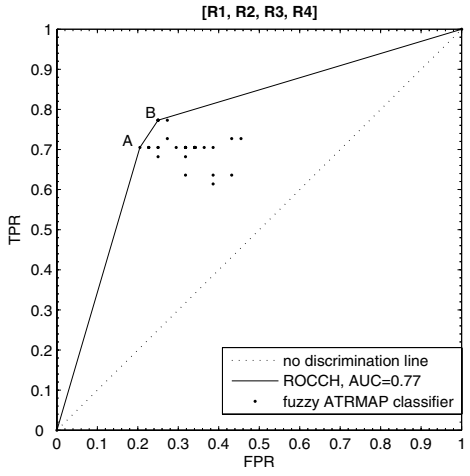


Fig. 6. ROC space for Fuzzy ARTMAP with four-ratio dataset. Point B of the ROCCH represents best classifier with vigilance parameter values from 0.6 to 0.675

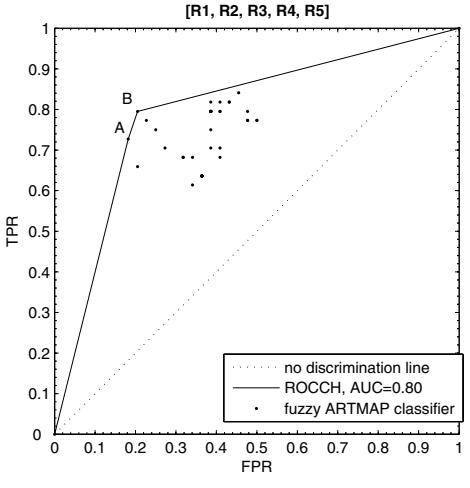


Fig. 7. ROC space for Fuzzy ARTMAP with five-ratio dataset. Point B of the ROCCH represents best classifier with vigilance parameter value 0.8.

'best' classifiers in the terms of ROC are the same best according to the accuracy analysis discussed above. Thus, we have a ROC confirmation of the choice of threshold value and that the ACC is a reliable figure of merit in our case.

A crisp or discrete classifier, such as Fuzzy ARTMAP NN plots a single point in the ROC space. Varying the vigilance parameter produces different classifiers, or aggregation of points in the ROC space. Figures 6 and 7 show ROC space of the Fuzzy ARTMAP NN with reduced and full variable sets respectively. In general,

given two points (classifiers), we can construct any 'intermediate' classifier just randomly weighting both classifiers (giving more or less weight to one or the other).

This creates a continuum of classifiers between any two classifiers, which allows linking of the two points by a line. Given several classifiers we can construct the ROC convex hull (ROCCH) curve connecting the most northwest points as well as the two trivial classifiers (0,0) and (1,1).

All classifiers below the ROCCH curve can be discarded because there is no combination of class distribution / cost matrix for which they could be optimal. Since only the classifiers on the ROCCH are potentially optimal, no others need be retained. This allows determining the candidates for optimal classifiers: points A and B for either case in figures 6 and 7. Analysis of results shows that in either case point B (most northwest) is optimal. In the case of reduced variable set B corresponds to vigilance parameter values between 0.6 and 0.675 and accuracy 76.1%; in the case of full variable set B corresponds to vigilance 0.8 and accuracy 79.5%. This again confirms that the choice of vigilance parameter value from the accuracy analysis is valid from point of view of ROC analysis.

5.3 AUC

In order to compare classifiers we had to reduce ROC performance to a single scalar value that represents expected performance. Three common approaches are in use: the intercept of the ROC curve with the line at 90 degrees to the no-discrimination line; the area between the ROC curve and the no-discrimination line; the area under the ROC curve, or "AUC". We adopted AUC as it has an important statistical property: the AUC of a classifier is equivalent to the probability that the classifier will rank a randomly chosen positive instance higher than a randomly chosen negative instance. This is equivalent to the Wilcoxon test of ranks. Since the AUC is a portion of the area of the unit square, its value will always be between 0.0 and 1.0. The AUC for useful classifiers is constrained between 0.5 (representing chance behavior) and 1.0 (representing perfect classification performance). The best classification model maximizes the AUC index. Calculation of AUC for crisp classifiers requires trapezoidal approximation by the formula:

$$AUC = \frac{1}{2} \sum_{i \in ROCCH} (TPR_i + TPR_{i+1})(FPR_{i+1} - FPR_i) \tag{11}$$

Calculations show that AUC_{BPNN2}=0.729; AUC_{BPNN3}= 0.697. This implies that a BPNN with 2 hidden nodes provides a better overall performance than a BPNN with 3 hidden nodes, although the accuracy of both at the optimal threshold value is the same. In order to compare overall performance of a Fuzzy ARTMAP with reduced and full set of variables, we calculated the area under the ROC convex hull. Results show that in the case of reduced set AUC=0.77; the full set yields AUC=0.80. These figures imply that using full set of Altman's financial ratios with Fuzzy ARTMAP NN provides better overall classification than using the reduced set and with certain vigilance parameter values it outperforms both BPNN and SOFM reaching accuracy of 79.5%.

5.4 Validation of Results

Cross-validation is a common way of measuring the error rate of a learning scheme on a particular dataset. We decided to use the leave-one-out cross-validation (LOOCV) procedure instead of dividing the dataset into training and testing datasets. This technique is suitable for small size datasets as it allows the greatest possible amount of data to be used for training. It is also a deterministic technique as no random sampling is involved, in contrast of a k-fold cross-validation (k<n). LOOCV is a n-fold cross-validation, where n is the number of instances in the dataset. Each instance in turn is left out, and the learning method is trained on all the remaining instances. The method is judged by its correctness on the remaining instance – one or zero for success or failure. The result of all n judgements, one for each instance is averaged in order to represents the final error estimate. All experiments and results discussed in this study were carried out according to the LOOCV procedure. This technique was also used in other studies, such as [9] and [6].

6 Conclusions

This study explores experimentally the potential of two types of neural networks, BPNNs and Fuzzy ARTMAP, to predict insolvency of Irish firms. We used financial information for a period of six years for a total of 88 Irish firms, represented as Altman's ratios and preprocessed in order to make data suitable for NN input. We also experimented with a reduced set of variables to see the effect of the curse of dimensionality and overfitting problems. Two architectures of BPNN were used – BPNN2 and BPNN3 with 2 and 3 hidden nodes respectively, as some heuristics suggests those two sizes. The NN performance was estimated by three metrics, accuracy, true positive rate, and false positive rate. The later two were involved in Receiver Operating Characteristics (ROC) analysis. Experiments show that BPNN2 and BPNN3 achieve the same best accuracy of 72.7%, but using different threshold functions. The ROC analysis confirms that those threshold functions are valid point of view of its metrics. AUC analysis shows that the overall performance of BPNN2 slightly exceeds that of BPNN3. Similar experiments were conducted with Fuzzy ARTMAP NN. Results show that this model outperforms both BPNN architectures with best accuracy of 79.5% at certain values of the vigilance parameter and full variable set. It outperforms also the self-organising feature maps (SOFM), which achieve 77.2% with the same dataset [9]. ROC analysis also confirms that those NN settings are best in the terms of the ROC metrics. Our experimental results were validated using the LOOCV technique. Another advantage of the Fuzzy ARTMAP model is that it provides fast one-pass online learning that retains already acquired knowledge, in contrast to the BPNN learning technique.

References

1. Altman, E.: Financial Ratios, Discriminant Analysis, and the Prediction of Corporate Bankruptcy. Journal of Finance 23(4), 598–609 (1968)
2. Atiya, A.: Bankruptcy Prediction for Credit Risk Using Neural Networks: A Survey and New Results. IEEE Transactions of Neural Networks 12(4), 929–935 (2001)

3. Balcaen, S., Ooghe, H.: 35 years of studies on business failure: An overview of the classical statistical methodologies and their related problems. Working paper 248, Ghent University, Belgium (2004)
4. Carpenter, G.A., Grossberg, S., Markuzon, N., Reynorlds, J.H., Rosen, D.B.: Fuzzy ARTMAP: A Neural Network Architecture for Incremental Supervised Learning of Analog Multidimensional Maps. IEEE Transaction on Neural Networks 3(5), 698–713 (1992)
5. Charalambous, C., Charitou, A., Kaourou, F.: Comparative analysis of artificial neural network models: application in bankruptcy prediction. Annals of Operations Research 99, 403–425 (2000)
6. Charitou, A., Neophytou, E., Charalambous, C.: Predicting Corporate Failure: Empirical Evidence for UK, European Acc. Review 13(3), 465–497 (2004)
7. Granger, E., Rubin, A., Grossberg, S., Lavoie, P.: A What-and-Where Fusion Neural Network for Recognition and Tracking of Multiple Radar Emitters. Neural Networks 3, 325–344 (2001)
8. Hagan, M.T., Menhaj, M.B.: Training feedforward networks with the Marquardt algorithm. IEEE Transactions on Neural Networks 5, 989–993 (1994)
9. Jones, M.: Financial Diagnosis of Irish Companies Using Self-Organising Neural Networks. In: Proceedings of the 9th Irish Academy of Management Annual Conference, Galway, Ireland, September, 7-9 (2005)
10. Kumar, P., Ravi, V.: Bankruptcy Prediction in Banks and Firms via Statistical and Intelligent Techniques. European Journal of Operational Research 180(1), 1–28 (2007)
11. Nachev, A.: Fuzzy ARTMAP neural network for classifying the financial health of a firm. In: Nguyen, N.T., et al. (eds.) IEA/AIE 2008. LNCS, vol. 5027, pp. 82–91. Springer, Heidelberg (2008)
12. Rees, W.: Financial Analysis. Prentice-Hall, Hemel Hempstead (1995)
13. Serrano-Cinca, C.: Self organizing neural networks for financial diagnosis. Decision Support Systems 17, 227–238 (1996)
14. Tan, Y., Van Cauwenberghe, A.: Neural-Network-Based D-Step-Ahead Predictors for Nonlinear Systems with Time Delay. Engineering Applications of Artificial Intelligence 12, 21–25 (1999)

Frequent Subgraph-Based Approach for Classifying Vietnamese Text Documents

Tu Anh Hoang Nguyen[1] and Kiem Hoang[2]

[1] Faculty of Information Technology, University of Science,VNU, Ho Chi Minh City, Vietnam
nhtanh@fit.hcmuns.edu.vn
[2] Faculty of Computer Science, University of Information Technology
VNU, Ho Chi Minh City, Vietnam
kiemhv@uit.edu.vn

Abstract. In this paper we present a simple approach for Vietnamese text classification without word segmentation, based on frequent subgraph mining techniques. A graph-based instead of traditional vector-based model is used for document representation. The classification model employs structural patterns (subgraphs) and Dice measure of similarity to identify a class of documents. This method is evaluated on Vietnamese data set for measuring classification accuracy. Results show that it can outperform k-NN algorithm (based on vector, hybrid document representation) in terms of accuracy and classification time.

Keywords: Text classification, Graph mining, Frequent subgraph, Vietnamese.

1 Introduction

In recent years we have seen a tremendous growth in the volume of text documents available on Internet. Technologies for efficient management of these documents are being developed continuously. One of representative tasks for efficient document management is Text Classification, also called as categorization. Automated text categorization is the task of assigning pre-defined class labels to incoming, unclassified documents [20]. It has numerous applications in fields like e-mail filtering, news monitoring, automated indexing of scientific articles, classification of news stories, searching for interesting information on WWW, etc.

Document representation model is one of the important factors involved in text classification. The popular model is to represent documents by keyword vectors according to standard vector space model with TF-IDF term weighting [16]. A number of classification algorithms have been introduced and operated on this model such as K-Nearest neighbor (k-NN) [11], Decision trees [1], Naïve Bayes [3], Neural network [23] and Support Vector machines (SVM) [6]. However, the traditional vector model suffers from fact that it loses important structural information in original text, such as the order in which terms appear or the locations of terms within the text.

In order to overcome the limitations of the vector space model, graph-based document representation model was introduced [15], [18]. The main benefit of graph-based techniques is that they allow us to keep inherent structural information of

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 299–308, 2009.
© Springer-Verlag Berlin Heidelberg 2009

original document. There are also mixed (hybrid) document models that use both representations: graph and vector. They were designed to overcome problems connected with simple representations. They capture structure information (by extracting relevant subgraphs) and represent relevant data using vector [9].

Automatic text classification of text documents in Asian languages is a challenging task. For Vietnamese text classification, it is necessary to cope with a problem called word segmentation since the language has no explicit word boundary delimiter. Vietnamese word segmentation itself is a difficult problem and affects to topic-based document classification.

In this paper we present a subgraph-based Vietnamese text classifier that overcomes not only the limitations associated with vector space model but also the effects of word segmentation problem. Instead of using word, we use "syllable" units, which are element linguistic units composing words to build graph. In this case, we can construct a graph that represents document without word segmentation step. We use frequent sub-graph mining techniques to identify representative features of a class (frequent sub-graphs) which is later used for classifying a new document. To the best of our knowledge this is the first time that graph model and graph mining techniques have been used for Vietnamese text.

The rest of paper is organized as follows: in section 2 we discuss related work, section 3 describes graph based document representation model, and section 4 gives detailed description of the classification procedure. Section 5 presents the classification results. Conclusions, and future work are in section 6.

2 Related Work

In this section we review several works done in text classification using graph representation model and methods have applied for Vietnamese text.

The authors of [15] used graph models for classifying web documents. The graph is constructed from text of a web page and words contained in the title and hyperlinks. An extension of k-NN algorithm with graph theoretical-distance measure based on maximal common subgraph was used to handle graph-based data. In [12], Galois lattice was used for discovering frequent subgraphs from document graphs and classification rules. An approach using hybrid document model was proposed in [9]. This method extracts subgraphs from a graph that represents web document then creates simple vector with Boolean values indicating relevant subgraphs. Popular algorithms such as k-NN, C4.5 and Naïve Bayes are used to build classifiers. They are reported to provide better results than methods using simple representations [10]. In [5], concepts of contrast and common subgraphs were combined with some ideas characteristic for emerging patterns technique to build a Contrast Common Patterns Classifier (CCPC). CPCC operates on a graph representation of a web document.

Previous works on Vietnamese text classification have used genetics algorithm-based classifier [8], Bayesian classification methods [17], Support Vector machines [21], and n-gram model [21] in developing automated text classification system.

Almost these systems use vector space model for document representation and required good word segmentation step.

3 Graph-Based Document Representation Model

In this section we present basic information on graph-based model for text document representation.

In graphic model, every document is transformed into a graph. There are numerous methods for creating graphs from documents. The authors of [15] described six major algorithms: standard, simple, n-distance, n-simple distance, absolute frequency and relative frequency. All these methods are based on adjacency of terms. Some of these methods were specially designed to deal with web documents by including markup elements information and some can also be used for plain (non-HTML) text document representation. In our case we used simple document representation because it most suitable for plain text.

The simple model is a labelled nodes directed graph $G = (V, E)$, where V is a set of *nodes*, E: is a collection of *edges*. In this model, each node v represents a unique term in a document. Each node is labelled with the term it represents. Multiple instances of a single term are counted as one unique node in the graph. Each edge e is an ordered pair of nodes (v_i, v_j). There will be an edge from v_i to v_j if, and only if, the term v_j appears successive to the term v_i in a document and the terms are not separated by certain punctuation marks (such as a period, comma).

The above definition of graph suggests that the number of nodes in a graph is the number of unique terms in a document.

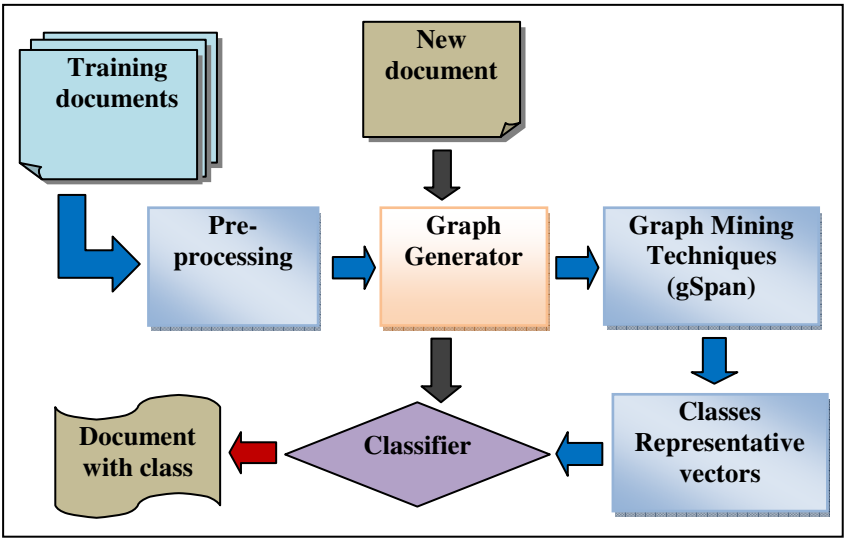


Fig. 1. ViCG - Text classification model

4 Vietnamese Text Classification Based on Frequent Subgraphs

We develop a text classification model called ViCG (*Vietnamese Classification based on Graph*) for Vietnamese documents. Our model uses graph - based representation and frequent subgraph discovery algorithm to find representative features of classes (or categories). In our approach, we consider text in a document as a concatenated sequence of syllables instead of words because we want to avoid Vietnamese word segmentation problem that is proved to be a very difficult problem. Once the complete set of frequent sub-graphs has been identified, we proceed to build representative vector of each class using this set. The overall flow is shown in Fig.1 and a brief description of each processing step is given below.

4.1 Pre-processing

All text documents go through a preprocessing stage. The preprocessing is performed for the documents to be classified and the training classes themselves. Pre-processing stage consists of the following steps:

- Convert text files to UTF-8 encoding.
- Standardize spelling: tone rule and letter variant processing.
- Transform the text document into a sequence of tokens consisting only morpho-syllables. This process is quite simple comparing to word segmentation algorithm.
- Remove stop words. We use maximum matching approach to remove stop words.
- Calculate the weight of morpho-syllables appearing in document. It is calculated similar as TF x IDF (term frequency x inverse document frequency) measure [16] in following way:

$$w(s_i, c_j) = SF_{i,j} \times \left(1 + \log \frac{N}{CF_i}\right) . \quad (1)$$

where $SF_{i,j}$ is the syllable frequency in the class c_j, N is the number of classes in the collection and CF_i is the number of classes containing the syllable s_i.

- Lastly, syllables are ranked based on their weight and only f % of most frequent syllables are retained for graph construction. This parameter is responsible for reducing computational complexity and is set to 80% currently. To construct the graph, only those syllables that are members of this frequent set are used.

4.2 Graph Construction

This step transforms Vietnamese text documents into graph format.

Graphs representing each document in the class are constructed from those syllables in the document that appear in the frequent set. We use “syllable” (as term) instead of “word” in forming nodes because Vietnamese words are usually composed of syllables and without word segmentation, the simple model can still capture

important structural relationships between words (or sets of words) within a document. A simple model that can be used for all type of text has been described in section 2. It is a directed graph; with chosen syllables from document form the nodes. Note that there is only a single node for each syllable even if a syllable appears more than once in a text. If syllable *x* and syllable *y* are adjacent in a document, then there is a directed edge from the node corresponding to *x* to the node corresponding to *y*. An edge is not added to the graph if the syllables are separated by certain punctuation marks (such as a period, comma).

4.3 Representative Feature Extraction

Graph mining aims at discovering interesting and representative patterns (substructures) within structural data. The authors of [7], [22], [24] give excellent review of graph mining algorithms.

During this step, graph mining techniques are used for extracting representative features. Document graphs are mined to discover the frequently occurring substructures (subgraphs). All directed graphs representing training documents are divided into disjoint classes. Then the frequent subgraph mining algorithm is activated on each class with a user-specified threshold value *minSup*. Only those subgraphs occurring in at least *minSup* % of the graphs are used to define representative features of class.

In our classification model, we extract frequently occurring subgraphs using the gSpan algorithm [19]. The gSpan (graph-based Substructure pattern mining) algorithm is a depth first search frequent subgraph miner. gSpan uses a canonical representation by mapping each graph to a unique code and uses depth-first search to discover frequent connected subgraphs without candidate generation. The original gSpan is designed for the search in undirected graph databases. We have done some minor changes in order to apply gSpan for set of directed graphs.

Next, we combine all representative features (frequent subgraphs in this case) discovered from each class to one features set. We build Boolean representative vector for particular class using this frequent subgraphs set (The *i*th entry of this vector is equal to "1" if frequent subgraph occurs in the representative features extracting from that class and otherwise). A set of representative binary vectors of all classes is the output of this stage.

4.4 Classification

The idea behind our classification model is very simple and effective.

Given a set of n documents $D = \{d_1, d_2, \ldots, d_n\}$, classified along a set C of m classes, $C = \{c_1, c_2, \ldots, c_m\}$. The representative of a particular class C_j is represented by a vector R_j with Boolean value indicating relevant frequent subgraphs. This lead to m representative vectors $\{R_1, R_2, ..R_m\}$, where each R_j is the representative for jth class. The class of the new document is determined as follows. First we use simple graph model to transform new document into graph G. After that, the graph G is mapped into the feature space of frequent subgraphs and represents new document as a binary vector x. The ith entry of this vector is "1" if the feature appears in the document's graph. The vector x of new document is compared to all m classes representative vectors in terms of similarity.

The new document will be classified as belonging to the class to whose representative it has the greatest Dice similarity [2].

$$\text{Dice}(x, R_j) = \frac{2|x \wedge R_j|}{|x| + |R_j|} . \quad (2)$$

where $|x|$, $|R_j|$ is the number of entries that are equal to "1" in vector x and R_j.

5 Experimental Evaluation

In order to evaluate the performance of our method, a corpus of Vietnamese text documents was built using news articles collected from online websites of several Vietnamese newspapers: VnExpress[1], TuoiTre Online[2], ThanhNien Online[3]. The corpus consists of text documents covering 7 categories: science, economy, health, sports, culture, informatics, and society and is described as follow. These documents are in size ranging from 1 KB to 15 KB.

Table 1. Detailed information on Vietnamese text document corpus

No	Category	No of documents
1	Science	358
2	Economy	654
3	Health	315
4	Sports	759
5	Culture	522
6	Informatics	457
7	Society	835
	Summary	3900

We used Precision, Recall and F_1 measures [14] to judge a classifier. Precision, Recall and F_1 are defined as:

$$\text{Recall} = \frac{A}{B} . \quad (3)$$

$$\text{Precision} = \frac{A}{C} . \quad (4)$$

$$F_1 = \frac{2 * \text{Recall} * \text{Presision}}{\text{Recall} + \text{Precision}} . \quad (5)$$

[1] http://www.vnexpress.net
[2] http://www.tuoitre.com.vn
[3] http://www.thanhnien.com.vn

where A is the number of documents correctly labeled as belonging to the category, B is the total number of documents that actually belong to the category, and C is the total number of documents labeled as belonging to the category.

The classification results are obtained by performing 5-way cross validation on our corpus. Table 2 shows the Recall, Precision, F_1 values, and their micro-averaging for all categories. The best result is for sports category with F_1 value of 0.921, and the worst result is for the health category with F_1 value of 0.739.

Table 2. Recall, Precision and F_1 performance of our method- ViCG

No	Category	Recall	Precision	F_1
1	Science	0.887	0.722	0.796
2	Economy	0.931	0.787	0.853
3	Health	0.639	0.875	0.739
4	Sports	0.873	0.968	0.921
5	Culture	0.798	0.941	0.864
6	Informatics	0.717	0.865	0.784
7	Society	0.792	0.933	0.857
	Micro-avg	**0.805**	**0.87**	**0.831**

In this work, we compare different classifiers based on different document representations. Table 3 shows comparison of best micro-average F_1 values for the following methods: k-NN with vector representation, Cosine similarity measure; k-NN with hybrid representation [9], and Manhattan similarity measure.

Table 3. Comparison of classification micro-average F_1

No	Method	Algorithm description	Micro-avg F_1
1	Vector	k-NN, Cosine similarity measure	0.708
2	Hybrid 1	k-NN, Manhattan similarity measure, "syllable" forms node	0.731
3	Hybrid 2	k-NN, Manhattan similarity measure, "word" forms node	0.716
4	*ViCG*	*Dice similarity measure, "syllable" forms node*	*0.831*

In k-NN model with vector representation, we use the state-of-the-art Vietnamese word segmentation program [4] and TF-IDF scheme. In k-NN with hybrid representation, we construct two kinds of document graphs: one with each node represents "syllable" (called as "Hybrid 1") and another with "word' forms the node (as "Hybrid 2"). In the type "Hybrid 2" of document graph, we use the same Vietnamese word segmentation program.

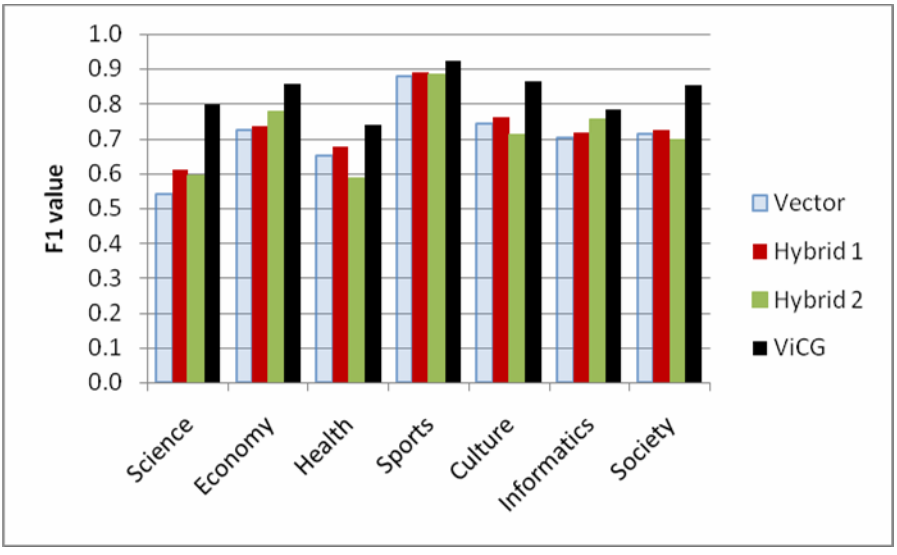


Fig. 2. Comparative F1 values for all categories

Figure 2 shows comparison of F_1 measure between our approach ViCG and others. ViCG outperforms other methods for this corpus.

6 Conclusions

In this paper, we propose an effective approach without word segmentation for classifying Vietnamese text documents using graph model and graph mining techniques. The results show that our method is competitive to existing schemes in term of accuracy. Furthermore construction and structure of our classifier is quite simple while avoiding word segmentation.

As for future research, some issues are still open:

- We need do more experiments with larger data to affirm the effectiveness of our method.
- Some techniques should be developed for finding the optimal representation model and discriminative classes' representative features.
- We need to adapt graph model to work with others classification algorithm like Naïve Bayes, SVM.

Acknowledgements. The authors would like to thank Prof. Dien Dinh and VCL group (Vietnamese Computational Linguistics) from the University of Science, VNU, HCM City for providing Vietnamese word segmentation tool. The authors also thank the anonymous reviewers for their helpful comments.

References

1. Apte, C., Damerau, F., Weiss, S.: Text mining with decision rules and decision trees. In: Proceedings of the Conference on Automated learning and discovery, Workshop 6: Learning from Text and the Web (1998)
2. Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval. Addison-Wesley, Reading (1999)
3. Baker, L.D., McCallum, A.K.: Distributional clustering of words for text classification. In: Proceedings of SIGIR, pp. 96–103 (1998)
4. Dien, D., Kiem, H., Toan, N.V.: Vietnamese Word Segmentation. In: Proceedings of 6th Natural Language Processing Pacific Rim Symposium, pp. 749–756 (2001)
5. Dominik, A., Walczak, Z., Wojceichowski, J.: Classification of web document using a graph-based model and structural patterns. In: Kok, J.N., Koronacki, J., Lopez de Mantaras, R., Matwin, S., Mladenič, D., Skowron, A. (eds.) PKDD 2007. LNCS (LNAI), vol. 4702, pp. 67–78. Springer, Heidelberg (2007)
6. Joachims, T.: Text categorization with support vector machines: Learning with many relevant features. In: Nédellec, C., Rouveirol, C. (eds.) ECML 1998. LNCS, vol. 1398, pp. 137–142. Springer, Heidelberg (1998)
7. Gudes, E., Shimony, S.E., Vanetik, N.: Discovering Frequent Graph Patterns using Disjoint Paths. IEEE Transaction on Knowledge and Data Engineering 18(11), 1441–1456 (2006)
8. Hung, N., Ha, N., Thuc, V., Nghia, T., Kiem, H.: Internet and Genetics Algorithm-based Text Categorization for Documents in Vietnamese. In: Proceedings of 3rd International Conference Research, Innovation and Vision of the Future, pp. 168–172 (2005)
9. Markov, A., Last, M.: A Simple, Structure-Sensitive Approach for Web Document Classification. In: Szczepaniak, P.S., Kacprzyk, J., Niewiadomski, A. (eds.) AWIC 2005. LNCS (LNAI), vol. 3528, pp. 293–298. Springer, Heidelberg (2005)
10. Markov, A., Last, M., Kandel, A.: Model-based classification of web documents represented by graphs. In: Proceedings of Workshop on Knowledge Discovery on the Web at KDD, pp. 31–38 (2006)
11. Masand, B., Linoff, G., Waltz, D.: Classifying news stories using memory based reasoning. In: Proceedings of SIGIR (1992)
12. Phuc, D.: Document classification using graph model, frequent sub-graphs and Galois lattice. In: Poster Proceedings of 4th International Conference on Computer Science - Research, Innovation and Vision of the Future, pp. 33–38 (2006)
13. Phuc, D., Phung, N.T.K.: Using Naïve Bayes Model and Natural Language Processing for Classifying Messages on Online Forum. In: Proceedings of IEEE International Conference on Computer Science - Research, Innovation and Vision for the Future, pp. 247–252 (2007)
14. Sebastiani, F.: Machine Learning in Automated Text Categorization. ACM Computing Surveys 34(1), 1–47 (2002)
15. Schenker, A., Last, M., Bunke, H., Kandel, A.: Classification Of Web Documents Using Graph Matching. International Journal of Pattern Recognition and Artificial Intelligence, Special Issue on Graph Matching in Computer Vision and Pattern Recognition 18(3), 475–479 (2004)
16. Salton, G., Wong, A., Yang, C.S.: A vector space model for automatic indexing. Communication of ACM 18(11), 613–620 (1992)

17. Thanh, V.N., Hoang, K.T., Thanh, T.T.N., Hung, N.: Word Segmentation for Vietnamese Text Categorization: An online corpus approach. In: Poster Proceedings of 4th International Conference on Computer Science - Research, Innovation and Vision for the Future, pp. 113–118 (2006)
18. Tomita, J., Nakawatase, H., Ishii, M.: Graph-based Text Database for Knowledge Discovery. In: Proceedings of 13th international World Wide Web conference on Alternate track papers & posters, pp. 454–455 (2004)
19. Yan, X., Han, J.: gSpan: Graph-Based Substructure Pattern Mining. In: Proceedings of 2002 IEEE International Conference on Data Mining, pp. 721–724 (2002)
20. Yang, Y., Liu, X.: A re-examination of text categorization methods. In: Proceedings of ACM SIGIR, pp. 42–49 (1999)
21. Vu, C.D.H., Dien, D., Nguyen, L.N., Hung, Q.N.: A Comparative Study on Vietnamese Text Classification Methods. In: Proceedings of IEEE International Conference on Computer Science - Research, Innovation and Vision for the Future, pp. 267–273 (2007)
22. Washito, T., Motoda, H.: State of the art of Graph-Based Data Mining. SIGKDD Exploration 5(1), 59–68 (2003)
23. Wiener, E., Pedersen, J.O., Weigend, A.S.: A neural network approach to topic spotting. In: Proceedings of 4th Annual Symposium on Document Analysis and Information Retrieval, pp. 317–332 (1995)
24. Worlein, M., Meinl, T., Fisher, I., Philippsen, M.: A quantative comparison of the subgraph miners MoFa, gSpan, FFSM, and Gaston. In: Jorge, A.M., Torgo, L., Brazdil, P.B., Camacho, R., Gama, J. (eds.) PKDD 2005. LNCS (LNAI), vol. 3721, pp. 392–403. Springer, Heidelberg (2005)

Random Projection Ensemble Classifiers

Alon Schclar and Lior Rokach

Department of Information System Engineering
and
Deutsche Telekom Research Laboratories
Ben-Gurion University, Beer-Sheva, Israel
{schclar,liorrk}@bgu.ac.il

Abstract. We introduce a novel ensemble model based on random projections. The contribution of using random projections is two-fold. First, the randomness provides the diversity which is required for the construction of an ensemble model. Second, random projections embed the original set into a space of lower dimension while preserving the dataset's geometrical structure to a given distortion. This reduces the computational complexity of the model construction as well as the complexity of the classification. Furthermore, dimensionality reduction removes noisy features from the data and also represents the information which is inherent in the raw data by using a small number of features. The noise removal increases the accuracy of the classifier.

The proposed scheme was tested using WEKA based procedures that were applied to 16 benchmark dataset from the UCI repository.

Keywords: Ensemble methods, Random projections, Classification, Pattern recognition.

1 Introduction

Ensemble methods are very popular tools in pattern recognition due to their robustness and higher accuracy relatively to non-ensemble methods [18]. Rather than relying on a single classifier, they incorporate several classifiers where, ideally, the combination of classifiers outperforms each of the individual classifiers. In fact, ensemble methodology imitates our second nature to seek several opinions before making any crucial decision. We weigh the individual opinions, and combine them before reaching a final decision [24].

Successful applications of the ensemble methodology can be found in many fields: finance [20], manufacturing [26] and medicine [22], to name a few.

One of the most common approaches for creating an ensemble classifier constructs multiple classifiers based upon a single given inducer e.g the nearest neighbor inducer and C4.5 [25]. The classifiers are constructed via a training step. Each classifier is trained on a different training set, all of which are derived from the original training set. The classification result of the ensemble algorithm combines the results of the different classifiers (e.g. by a voting scheme). Ensemble methods can also be applied to regressors in which case a multivariate function is used to combine the individual regression results of the classifiers [27]. In this paper, we focus on classifiers rather than on regressors.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 309–316, 2009.
© Springer-Verlag Berlin Heidelberg 2009

Two crucial components in an effective ensemble method are *accuracy* and *diversity*. Accuracy requires that each individual classifier will generalize as much as possible to new test instances i.e. *individually* minimize the generalization error. Diversity [19] requires that the individual generalization errors will be uncorrelated as much as possible. These components are contradictory in nature. On one hand, if every individual classifier in completely accurate, then the ensemble is not diverse and is not required at all. On the other hand, if the ensemble is completely diverse the ensemble classification is equivalent to random classification. In [23], "kappa-error" diagrams are introduced in order to show the effect of diversity at the expense of reduced individual accuracy.

When using classifiers that are derived from a single inducer, the diversity is achieved by construction of different training sets. One of the most common ensemble methods of this type is the Bagging algorithm [5] which obtains the diversity by creating the various training sets via bootstrap sampling (allowing repetitions) of the original dataset. This method is simple yet effective. Bagging was successfully applied to a variety of problems e.g. spam detection [30] and analysis of gene expressions [28].

In this paper, we utilize random projections to construct a novel ensemble algorithm. Specifically, a set of random matrices is generated. The training sets of the different classifiers are constructed by projecting the original training set onto the random matrices. This approach is different from the random subspaces [16] method that is used in [27]. In random subspace, each training set is composed of a random subset of *features*. However, in random projection, every derived feature is a random linear combination of the original features. In this sense, random subspaces are equivalent to random feature *selection* while random projections are equivalent to random feature *extraction*.

When designing the proposed algorithm, we aimed to construct an algorithm which will require limited computational resources i.e. the algorithm was designed so its complexity would be as low as possible. Accordingly, it stands to reason to compare the proposed algorithm only with algorithms of the same complexity category. The most prominent algorithm in this complexity category is the Bagging algorithm, whose complexity is only slightly lower than the complexity of the proposed algorithm. No comparison is made with more complex ensemble algorithms such as AdaBoost [12] and Rotation Forests [1].

1.1 Random Projections

The utilization of random projections as a tool for dimensionality reduction stems from the pioneering work of Johnson and Lindenstrauss [17] who laid the theoretical foundations of dimensionality reduction by proving its feasibility. Specifically, they showed that N points in N dimensional space can almost always be projected onto a space of dimension $C \log N$ with control on the ratio of distances and the error (distortion). Bourgain [4] showed that any metric space with N points can be embedded by a bi-Lipschitz map into an Euclidean space of $\log N$ dimension with a bi-Lipschitz constant of $\log N$. Thus, random projections reduce the dimensionality of a dataset while preserving its geometrical structure.

Applications of the above theorems, which use random projections for dimensionality reduction, were successfully used for protein mapping [21], reconstruction of frequency-sparse signals [7,9], face recognition [13] and textual and visual information retrieval [3].

Random projections were also utilized as part of an ensemble algorithm for clustering in [10] and for gene expression data analysis in [11]. Essentially, the random projection algorithm is used to reduce the dimensionality of the dataset. Then an EM (of Gaussian mixtures) clustering algorithm is applied to the dimension-reduced data. However, it is reported in [10] that using a single run of the random projection algorithm produces poor and unstable results. This is due the unstable nature of random projections. Accordingly, an ensemble algorithm is proposed. Each iteration in the algorithm is composed of two steps: (a) dimensionality reduction via random projection and (b) application of the EM clustering algorithm. The ensemble algorithm achieves results that are better and more robust than those obtained by single runs of random projection/clustering and are also superior to a similar scheme which uses PCA to reduce the dimensionality of the data.

In the following, we formally describe the random projection algorithm for dimensionality reduction. Let

$$\Gamma = \{x_i\}_{i=1}^{N} \tag{1}$$

be the original high-dimensional dataset given as a set of column vectors where $x_i \in \mathbb{R}^n$, n is the (high) dimension and N is the size of the dataset. All dimensionality reduction methods embed the vectors into a lower dimensional space $\mathbb{R}^q$ where $q \ll n$. Their output is a set of column vectors in the lower dimensional space

$$\widetilde{\Gamma} = \{\widetilde{x}_i\}_{i=1}^{N}, \ \widetilde{x}_i \in \mathbb{R}^q \tag{2}$$

where q approximates the intrinsic dimensionality of Γ [15,14]. We refer to the vectors in the set $\widetilde{\Gamma}$ as the *embedding vectors*.

In order to reduce the dimensionality of Γ using random projections, a random vector set $\Upsilon = \{r_i\}_{i=1}^{n}$ is first generated where $r_i \in \mathbb{R}^q$. Two common choices for generating the random basis are:

1. The vectors $\{r_i\}_{i=1}^{n}$ are uniformly (or normally) distributed over the q dimensional unit sphere.
2. The elements of the vectors $\{r_i\}_{i=1}^{n}$ are chosen from a Bernoulli +1/-1 distribution and the vectors are normalized so that $\|r_i\|_{l_2} = 1$ for $i = 1, \ldots, n$.

Next, a $q \times n$ matrix R whose columns are composed of the vectors in Υ, is constructed. The embedding $\widetilde{x}_i$ of x_i is obtained by

$$\widetilde{x}_i = R \cdot x_i$$

2 The Proposed Algorithm

In the proposed algorithm, random projections are used in order to create the training sets on which the classifiers will be trained. Using random projections provides the required diversity component of the ensemble method. Although the complexity of using random projections is slightly higher than that of the Bagging algorithm, random projections possess useful properties that can help obtain better classification results than those achieved by the Bagging algorithm. In particular, random projections reduce

the dimensionality of the dataset while maintaining its geometrical structure within a certain distortion rate [17,4]. This reduces the complexity of the classifier construction as well as the complexity of the classification of new members while producing classifications that are close to or better than those of the original dataset. Furthermore, dimensionality reduction removes noisy features and thus can improve the generalization error.

One of the crucial parameters to any dimensionality reduction algorithm is the dimension of the target space. In the proposed algorithm, we set the dimension of the target space to a portion of the dimension of the ambient space where the training members reside.

Algorithm Description. Given a training set Γ as described in Eq. 1, we construct a matrix G of size $n \times N$ whose columns are composed of the column vectors in Γ

$$G = (x_1|x_2|\ldots|x_N)$$

Next, we generate k random matrices $\{R_i\}_{i=1}^{k}$ of size $q \times n$ where q and n are described in the previous section and k is the number of classifiers in the ensemble. The columns are normalized so that their l_2 norm will be 1.

The training sets $\{T_i\}_{i=1}^{k}$ for the ensemble classifiers are constructed by projecting G onto the random matrices $\{R_i\}_{i=1}^{k}$, i.e. $T_i = R_i \cdot G$ where $i = 1, \ldots, k$. These training sets are input to an inducer $\mathfrak{I}$ and the outcome in a set of classifiers $\{\mathfrak{C}_i\}_{i=1}^{k}$.

In order to classify a new member u by a classifier $\mathfrak{C}_i$, u must first be embedded into the dimension-reduced space $\mathbb{R}^q$. This is achieved by projecting u onto the random matrix R_i

$$\tilde{u} = R_i \cdot u.$$

where $\tilde{u}$ is the embedding of u. The classification of u is set to the classification of $\tilde{u}$ by $\mathfrak{C}_i$. The final classification of $\tilde{u}$ by the proposed ensemble algorithm is produced via a voting scheme that is applied to the classification outcomes of all the classifiers $\{\mathfrak{C}_i\}_{i=1}^{k}$ for the $\tilde{u}$.

3 Experimental Results

We tested our approach on 16 datasets from the UCI repository [2] which contains commonly used benchmark datasets that are used to test machine learning algorithms e.g. classifiers. We used the nearest-neighbor inducer (WEKA's B1 lazy classifier) to construct 10 classifiers in each ensemble where the results are the average of 10 ensembles. The size of the dimension-reduced space was set to half of the dimension of the training set. The random matrices were generated from a Uniform distribution.

Table 1 describes the results of the experiments comparing the performance of the proposed algorithm with the performance of the Bagging algorithm. We also include the results of the simple non-ensemble nearest-neighbor (NENN) classifier and an ensemble algorithm which is based on the Random subspaces (RS) approach (each training set contains 50 percent randomly chosen features). For each dataset, we specify the number of instances, the number of features (original dimensionality) and the *Generalized*

Table 1. Properties of the benchmark datasets along with a comparison between the performance of the proposed algorithm and the Bagging algorithm. The '++' postfix means that the proposed algorithm is significantly more accurate than the Bagging algorithm. The converse is marked by a '–' postfix. The '+' postfix indicates that the proposed algorithm is more accurate than the Bagging algorithm without statistical significance. The two right columns contain the results of a Random-subspace based ensemble algorithm and a non-ensemble nearest neighbor classification algorithm.

Dataset Name	Instance#	Feature#	Proposed algorithm	Bagging	Random subspaces	Non-ensemble NN
Hill Valley ++	2424	100	73.15±7.41	61.38±5.09	61.89±4.11	61.38±4.3
Isolet ++	7797	617	90.56±1.02	89.77±1.02	90.01±1.03	87.09±1.19
Madelon ++	2000	500	67.72±3.36	55.63±3.29	55.1±3.47	53.25±3.13
Multiple features –	2000	649	96.11±1.3	97.9±0.9	97.9±0.92	97.65±1.01
Sat ++	6435	36	91.06±1.06	90.41±0.92	90.41±0.97	88.97±1.12
Segment –	2310	19	96.27±1.21	97.03±1.17	97.15±1.11	96.76±1.1
Shuttle –	58000	9	99.79±0.06	99.93±0.03	99.93±0.03	99.75±0.06
Spambase –	4601	57	85.56±1.44	91±1.35	90.78±1.36	86.56±1.71
Waveform w noise ++	5000	40	79.63±1.88	73.8±1.7	73.41±1.82	73.22±2.13
Waveform w/o noise ++	5000	21	80.93±1.81	77.4±1.67	77.17±1.63	71.91±1.88
Wine –	178	13	76.96±8.87	95.07±4.31	95.12±4.34	91.07±6.12
Musk1 +	476	166	86.98±4.73	85.65±4.91	85.55±4.79	83.51±5.27
Musk2 +	6598	166	95.89±0.67	95.79±0.7	95.7±0.72	95.43±0.72
Ecoli +	336	7	83.42±5.38	80.98±6.1	80.66±6.16	73.96±6.15
Glass +	214	9	71.44±9.18	69.98±9.25	70.3±8.96	73.03±9.95
Ionosphere +	351	34	90.4±4.55	87.36±5.06	87.1±5.12	85.62±5.21

Accuracy of the two algorithms. The generalized accuracy represents the mean probability that an instance was classified correctly and it was calculated via a 10-fold cross-validation procedure which was repeated ten times. Since the average accuracy is a random variable, the confidence interval was estimated by using the normal approximation of the binomial distribution. The one-tailed paired t-test [8] with a confidence level of 95% verified whether the differences in accuracy between the proposed algorithm and the Bagging algorithm were statistically significant. It can be seen in Table 1 that the proposed algorithm significantly outperforms the Bagging algorithm in six (Hill Valley, Isolet, Madelon, Sat, Waveform with noise and Waveform without noise) datasets out of the 16 benchmark datasets. Furthermore, the proposed algorithm outperforms without statistical significance the Bagging algorithm in five (Musk1, Musk2, Ecoli, Glass, Ionosphere) out of the 16 benchmark datasets. On the other hand, the Bagging algorithm significantly outperforms the proposed algorithm in five datasets (Multiple features, Segment, Shuttle, Spambase and Wine). However, the dimensionality in these cases is less than 101 and the proposed algorithm dominates the datasets whose dimension is higher than 100. The proposed algorithm outperforms each of the NENN and the RS algorithms in 11 of the test datasets. The NENN algorithm and the RS algorithm outperform the proposed algorithm in five datasets. In three of which (Segment, Multiple features and Wine), both of them outperform it.

In order to conclude which algorithm performs best over all the benchmark datasets, we use the Wilcoxon test [8] whose definition follows: let δ_i be the difference between the performance scores of the two classifiers on the j-th out of the $\Omega = 16$ datasets. We rank the differences according to their absolute values. In case of ties, average ranks are assigned. Let ρ^+ be the sum of ranks for the data sets on which the proposed algorithm outperformed the Bagging algorithm, and ρ^- be the sum of ranks for the opposite.

Cases for which $\delta_i = 0$ are split evenly between the sums. Formally, $\rho^+$and ρ^- are defined as follows:

$$\rho^+ = \sum_{\delta_i > 0} rank\left(\delta_i\right) + \frac{1}{2}\sum_{\delta_i = 0} rank\left(\delta_i\right);$$

$$\rho^- = \sum_{\delta_i < 0} rank\left(\delta_i\right) + \frac{1}{2}\sum_{\delta_i = 0} rank\left(\delta_i\right)$$

Let $\tau = \min\left(\rho^+, \rho^-\right)$ be the smaller of the sums. Define the statistic

$$z = \frac{\tau - \frac{1}{4}\Omega\left(\Omega + 1\right)}{\sqrt{\frac{1}{24}\Omega\left(\Omega + 1\right)\left(2\Omega + 1\right)}}$$

which for a larger number of data sets is distributed approximately normally. For the datasets that we used, we got $z = -1.29$ to $\alpha = 0.1$. Thus, the proposed algorithm significantly outperforms the Bagging algorithm with $z = -1.29, p < 0.1$.

4 Conclusions and Future Work

In the this paper, we introduced an alternative ensemble method to the Bagging algorithm. The method uses random projections instead of the bootstrap sampling that is used by the Bagging algorithm. The proposed method proves to be superior to the bagging algorithm in several datasets while producing competitive results for the other datasets.

The results in this paper are promising. However, a question that needs further investigation is when does the proposed method outperform the Bagging algorithm. Ideally, rigorous criteria should be formulated.

Furthermore, the proposed method should also be tested using other inducers e.g. classification and regression trees [6], SVM [29], etc.

References

1. Alonso, C.J.: Rotation forest: A new classifier ensemble method. IEEE Transactions on Pattern Analysis and Machine Intelligence 28(10), 1619–1630 (2006)
2. Asuncion, A., Newman, D.J.: UCI machine learning repository (2007)
3. Bingham, E., Mannila, H.: Random projection in dimensionality reduction: applications to image and text data. In: Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2001), San Francisco, CA, USA, August 26-29, 2001, pp. 245–250 (2001)
4. Bourgain, J.: On lipschitz embedding of finite metric spaces in Hilbert space. Israel Journal of Mathematics 52, 46–52 (1985)
5. Breiman, L.: Bagging predictors. Machine Learning 24(2), 123–140 (1996)

6. Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J.: Classification and Regression Trees. Chapman & Hall, Inc., New York (1993)
7. Candès, E., Romberg, J., Tao, T.: Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. IEEE Transactions on Information Theory 52(2), 489–509 (2006)
8. Demsar, J.: Statistical comparisons of classifiers over multiple data sets. Journal of Machine Learning Research 7, 1–30 (2006)
9. Donoho, D.L.: Compressed sensing. IEEE Transactions on Information Theory 52(4), 1289–1306 (2006)
10. Zhang Fern, X., Brodley, C.E.: Random projection for high dimensional data clustering: A cluster ensemble approach, pp. 186–193 (2003)
11. Folgieri, R.: Ensembles based on Random Projection for gene expression data analysis. PhD thesis, University of Milano (2007)
12. Freund, Y., Schapire, R.: Experiments with a new boosting algorithm. machine learning. In: Proceedings for the Thirteenth International Conference, pp. 148–156. Morgan Kaufmann, San Francisco (1996)
13. Goel, N., Bebis, G., Nefian, A.: Face recognition experiments with random projection. In: Proceedings of SPIE, vol. 5779, p. 426 (2005)
14. Hegde, C., Wakin, M., Baraniuk, R.G.: Random projections for manifold learning. In: Neural Information Processing Systems (NIPS) (December 2007)
15. Hein, M., Audibert, Y.: Intrinsic dimensionality estimation of submanifolds in Euclidean space. In: Proceedings of the 22nd International Conference on Machine Learning, pp. 289–296 (2005)
16. Ho, T.: The random subspace method for constructing decision forests. IEEE Transactions on Pattern Analysis and Machine Intelligence 20, 832–844 (1998)
17. Johnson, W.B., Lindenstrauss, J.: Extensions of Lipshitz mapping into Hilbert space. Contemporary Mathematics 26, 189–206 (1984)
18. Kuncheva, L.I.: Combining Pattern Classifiers. Methods and Algorithms. John Wiley and Sons, Chichester (2004)
19. Kuncheva, L.I.: Diversity in multiple classifier systems (editorial). Information Fusion 6(1), 3–4 (2004)
20. Leigh, W., Purvis, R., Ragusa, J.M.: Forecasting the nyse composite index with technical analysis, pattern recognizer, neural networks, and genetic algorithm: a case study in romantic decision support. Decision Support Systems 32(4), 361–377 (2002)
21. Linial, M., Linial, N., Tishby, N., Yona, G.: Global self-organization of all known protein sequences reveals inherent biological signatures. Journal of Molecular Biology 268(2), 539–556 (1997)
22. Mangiameli, P., West, D., Rampal, R.: Model selection for medical diagnosis decision support systems. Decision Support Systems 36(3), 247–259 (2004)
23. Margineantu, D.D., Dietterich, T.G.: Pruning adaptive boosting. In: Proceedings of the 14th International Conference on Machine Learning, pp. 211–218 (1997)
24. Polikar, R.: Ensemble based systems in decision making. IEEE Circuits and Systems Magazine 6(3), 21–45 (2006)
25. Quinlan, R.R.: C4.5: programs for machine learning. Morgan Kaufmann Publishers Inc., San Francisco (1993)
26. Rokach, L.: Mining manufacturing data using genetic algorithm-based feature set decomposition. International Journal of Intelligent Systems Technologies and Applications 4(1/2), 57–78 (2008)

27. Rooney, N., Patterson, D., Tsymbal, A., Anand, S.: Random subspacing for regression ensembles. Technical report, Department of Computer Science, Trinity College Dublin, Ireland, February 10 (2004)
28. Valentini, G., Muselli, M., Ruffino, F.: Bagged ensembles of svms for gene expression data analysis. In: Proceeding of the International Joint Conference on Neural Networks - IJCNN, pp. 1844–1849. IEEE Computer Society Press, Los Alamitos (2003)
29. Vapnik, V.N.: The Nature of Statistical Learning Theory (Information Science and Statistics). Springer, Heidelberg (1999)
30. Yang, Z., Nie, X., Xu, W., Guo, J.: An approach to spam detection by naive bayes ensemble based on decision induction. In: Proceedings of the Sixth International Conference on Intelligent Systems Design and Applications (ISDA 2006) (2006)

Knowledge Reuse in Data Mining Projects and Its Practical Applications

Rodrigo Cunha, Paulo Adeodato, and Silvio Meira

Center of Informatics, Federal University of Pernambuco
Caixa Postal 7851, Cidade Universitária, 50732-970, Recife-PE, Brazil
{rclvc,pjla,srlm}@cin.ufpe.br

Abstract. The objective of this paper is providing an integrated environment for knowledge reuse in KDD, for preventing recurrence of known errors and reinforcing project successes, based on previous experience. It combines methodologies from project management, data warehousing, mining and knowledge representation. Different from purely algorithmic papers, this one focuses on performance metrics used for managerial such as the time taken for solution development, the amount of files not automatically managed and other, while preserving equivalent performance on the technical solution quality metrics. This environment has been validated with metadata collected from previous KDD projects developed and deployed for real world applications by the development team members. The case study carried out in actual contracted projects have shown that this environment assesses the risk of failure for new projects, controls and documents all the KDD project development process and helps understanding the conditions that lead KDD projects to success or failure.

Keywords: Data mining project, Knowledge reuse in KDD projects, Risk assessment of KDD projects.

1 Introduction

Early research on artificial intelligence (AI) focused on the implementation and optimization of algorithms. These algorithms however, only produced reliable results in very specific applications, in limited domains. The general application of AI to data generating real world activities, data mining, was far from satisfactory, mainly due to the difficult integration of the databases, to the low quality of the data available and to the poor understanding of the business operation (application domain). In 1996, Fayyad et al. [1] generalized the scope inserting data mining in a more global process coined Knowledge Discovery in Databases (KDD). Also in 1996, potential data mining solutions' consumers and suppliers formed a consortium for creating a methodology for systematically developing data mining solutions for real problems. They came up with the CRISP-DM (Cross-Industry Standard Process for Data Mining) [2], a non-proprietary methodology for identifying and decomposing a data mining project in several stages, shared by all domains of application. Those initiatives aimed at standardizing the development process of data mining solutions

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 317–324, 2009.
© Springer-Verlag Berlin Heidelberg 2009

which involves the use of several tools for modeling, data visualization, analysis and transformation, performance evaluation and even specific programming tasks. Once this standard had been created, the provision of interoperability among the several different platforms in a single environment where all the processes are centralized and documented became one of the most important issues in KDD applications to real world problems.

According to Bartlmae and Riemenschneider [3], another important issue in KDD projects nowadays, mainly due to their complexity and strong user dependence, is the inadequate documentation, management and control of the experiences in solution development, thus yielding the recurrence of errors already known from previous projects in new ones. The lack of a platform capable of reusing the knowledge and lessons learned in previous projects' developments is a practical problem worsened by the inadequate interoperability among the data mining tools available in the platforms for KDD [4]. Summarizing, the lack of proper interoperability together with the lack of knowledge reuse capability in KDD solution development platforms are deficiencies that may lead projects to failure or delays and cause client dissatisfaction and cost increase.

This paper presents the capability of knowledge reuse from previous data mining projects. Therefore, this environment provides a better understanding of the conditions which make KDD projects turn into a failure or a success and a simpler and more precise parameter specification for producing high quality KDD projects that match the clients' expectations within the schedule and budget planned.

This paper is organized as follows. Section 2 presents the literature survey on approaches related to the proposed environment. Section 3 describes the architecture and functionality of the Knowledge Reuse Environment. Section 4 shows the relevant results for knowledge reuse in data mining projects. Finally, Section 5 summarizes the research carried out, emphasizes its main results along with their interpretation, states its limitations, and proposes future work for improving the Knowledge Reuse.

2 Literature Review

IMACS (Interactive Market Analysis and Classification System) [5] was one of the first initiatives to consider involving the user in the KDD process, back in 1993. When that system was first proposed, data mining tools used to provide very limited functionality and IMACS' development has not followed the evolution of those tools. Thus, IMACS provides support for only the creation of semantic definitions for the data and the formal representation of knowledge. In 1997, CITRUS [6] was proposed based on the CRISP-DM methodology. Two years later, UGM (User Guidance Modulates) was presented [7] as an improvement to CITRUS based on experiences of past projects for knowledge reuse. In 2002, IDAs [8] was proposed based on Fayyad et al.'s methodology. After the Clementine release, the vast majority of tools (market tools and academic tools), adopted an interface focused on the data mining workflow.

Other relevant work is the application of Case Based Reasoning for Knowledge Management in KDD Projects [3], which proposes a environment aimed at reusing knowledge in data mining projects. The idea is based on the concept of Experiences

Factory where Case Based Reasoning (CBR) helps storing and retrieving knowledge packages in a data repository.

The Statlog project [9] proposes a methodology to evaluate the performance of different algorithms for machine learning, neural networks and statistics. In spite of using the knowledge reuse concept, the scope of reuse in Statlog is limited to data mining algorithms.

Finally, the environment for Distributed Knowledge Discovery Systems project [10] introduces a environment aimed at integrating different data mining tools and platforms related to organizational modeling and integration of solutions. Despite considering integration an important issue, this approach only integrates the data mining tools. It does not deal with either metadata acquisition or meta-data mining on stored knowledge.

In summary, the literature offers isolated initiatives for knowledge reuse, but no contribution considering both of these features with focus on the process, as presented in this paper.

3 Knowledge Reuse Environment

In this environment, the knowledge databases are stored in three different structures, according to the types of their contents.

1) Metadata Database: This database stores information from previous projects. Here, metadata are all types of information produced along the KDD project development process, such as: data transformation needed, algorithms used, number of components used, project manager, project duration, overall project cost and client's level of satisfaction among others. That is, the metadata database stores information ranging from project management to specific algorithms with their corresponding performances. This module consists of two sub-modules: the transactional metadata database and the managerial metadata database. The transactional metadata database stores all of the project's metadata in a relational logic model. Managerial metadata databases are constructed via an ETL (Extract, Transform, and Load) process [11] carried out on the transactional database. These managerial metadata databases, also called Data Marts, are represented in a star model [11]. The objective of these metadata databases is providing support for both the project manager and the KDD experts along all the KDD solution development process.

2) CBR Projects: This module stores the knowledge of past projects through the technique of Case Based Reasoning (CBR) [3]. The purpose of this module is to reuse cases similar to the current project (being or to be developed) for providing the data mining expert with the adequate condition for making the best decisions in the new project. Currently, the environment offers three milestones for decision support. In the first milestone, it helps the project manager estimate the risk of the project being a success or a failure, even before it has started, and recovers the most similar cases to the new project. The second milestone occurs at the projects' planning stages where the goal is to define the most appropriate data mining tasks (classification, forecasting etc) based on the most similar past projects. Finally, at the third milestone, available at the preprocessing stage, the environment analyzes and extracts the most similar past transformations of the data.

3) Learned Lessons Database: This module stores the lessons of previous projects through the technique of Case Based Reasoning. Learned Lessons consist of problems, solutions, suggestions and observations that the experts have catalogued in previous projects with the objective of sharing them in future projects or training. In short, a learned lesson is an entry in the environment's module that makes the experience lived and catalogued by users available for future use. An example of an actual catalogued lesson learned refers to importing data in text file format into the SPSS (Statistical Package for The Social Sciences). This tool is likely to modify the formatting of numeric variables and truncate those of the string type. Now, the environment gives a warning for this problem in projects involving text file inputs to SPSS.

4 Knowledge Reuse and Experimental Platform

The knowledge stored can be reused in several ways, from supporting the decision of starting a new project to defining the most appropriate data transformation technique. This Section presents how it has been used and the results achieved in actual projects.

4.1 Problem Characterization

Here, the decision support system helps decide if a new data mining project should be developed or not, based on previous projects experience. Even before a new project starts, the system estimates its risk of failure (the higher the score, the higher the risk). If the risk is acceptable, the project starts; otherwise, the system presents the conditions that make the project risky for supporting project renegotiation or, in extreme cases, even project halt. This system helps saving a lot of money and time spent in re-work on ill specified projects.

A database collected along recent years of data mining project development by NeuroTech has been used for the environment performance assessment in an actual problem of meta-data mining. The metadata database has been imported from 69 data mining projects executed in the past; 27 labeled as "success" and 42 labeled as "failure" (69=27+42). The following three criteria were used for this labeling of project target classes: 1) The contracting client's evaluation (satisfied or dissatisfied); 2) NeuroTech's technical team evaluation: success or failure; and 3) Cost/benefit ratio resulting from the project: success or failure. When a project had a "negative" evaluation in any of the three criteria, it was labeled a failure; otherwise, it was labeled a success, in this binary classification modeling.

Each row of the metadata database represents a project developed whose metadata attributes are stored in its columns. For all projects, there are 19 input attributes (explanatory variables) and an output attribute (dependent variable) which represents the target class label (success or failure). Some of the explanatory variables were: company's (client) size (based on revenue), company's (client) experience with previous DW or KDD (number of projects developed) and if the present project needs behavioral data as input information among other variables. Logistic regression from Weka has been the statistical inference technique used for project risk estimation. Due to the small amount of labeled examples (69) available for modeling, the leave-one-out method has been applied as experimental data sampling strategy using MatLab code.

The technical performance evaluation of the system was assessed using the R-Project software in two distinct forms: 1) Separability between the distributions of successes and failures measured by the KS statistical test; 2) Simulation of several decision thresholds scenarios on the project scores produced.

4.1.1 Experimental Results on Risk Assessment

The quality of the meta-data mining is assessed by the usual data mining performance metrics. The performance achieved via leave-one-out reached a maximum value of 0.65 on the KS statistical test [12] which represents a statistically significant difference at α=0.05. This shows that, technically, the system can be used for decision support.

For finer decisions, Table 1 presents the scenario for several different score thresholds. For each threshold, it presents the rate of detection of failure in the projects, showing that higher score bands contain higher percentage of failures. New projects that produce scores above 75, for instance, have very high risk of failure and should be renegotiated for risk reduction, before the project start.

Table 1. Decision scenario for several score thresholds

Score band	Failures	Successes	Total
0-25	9 (30%)	21 (70%)	30
25-50	7 (64%)	4 (36%)	11
50-75	3 (60%)	2 (40%)	5
75-100	23(100%)	0 (0%)	23
Total	42 (61%)	27 (39%)	69

Should such a system be available for assessing the risk of these 69 projects in the past, just signaling those with scores above 75 would have prevented 23 out of the 42 failed projects without increased attention on any successful project. That would have represented a detection of 55% of the failures, from the start. As previously stated, this is an important managerial metrics for this paper.

4.1.2 CBR Measurements on Project Similarity

The same 69 projects used in the metadata database application were imported to the CBR Project database. In practice, the CBR implementation complements the logistic regression project, returning the cases most similar to the new project. In the end, the project manager has a score for project risk assessment and a collection of the most similar previous projects for decision support. For the cases' representation, Case Based Reasoning (CBR) with attribute-value representation [3] was the technique used. The similarity is divided into global similarity and local similarity. The global similarity is weighed and normalized nearest neighbour [13]. The local similarity is related to the attributes that describe the case, in other words, the local similarity depends on the nature of the attribute (string, binary, numeric and ordinal). For each attribute of the "string" type a similarity matrix was constructed by interviewing three NeuroTech's project managers. According to the opinion of each one an average opinion was inferred. For the ordinal and binary attributes, local similarity was

defined as the module difference of each attribute's values. For the numeric attributes the local similarity was defined by a linear function. In this case, the similarity grows as the weighed distance decreases. Once the structure of the cases and the similarity measures are defined, the CBR problem becomes the recovery of cases in the knowledge database. The recovery process is constituted of a group of sub-tasks. The first task is the assessment of the situation via a query through a group of relevant attributes. The second sub-task for case recovery is the matching strategy and selection. The objective is to identify a group of cases similar to that in query Q which returns k the most similar cases. In this work, the threshold was defined empirically as 0.5 similarity, Therefore only cases with similarity greater than or equal to 50% in relation to question Q will be returned

From the results achieved, NeuroTech decided to adopt the environment to estimate the failure probability of its projects using logistic regression and to find the most similar previously developed projects using CBR. Now, new projects go through the model assessment in order to estimate the chance their success before their development. The score threshold was defined as 75, i.e., only the projects with score below 75 will be automatically approved. Every project with a score higher than 75 should be evaluated by the company's committee, formed by the managers in charge of the business area, the customer area, and by the company's chief-scientist.

Only after the committee's approval, the project starts; otherwise, some contractual condition and/or project parameters should be altered based on similar cases and again submitted to the model for risk assessment.

Some subjective results have been achieved in NeuroTech with the use of the environment. For instance, a new project contracted by a retail business company for credit scoring solution was evaluated with an 89% chance of failure. When the NeuroTech operation manager used the environment for searching similar cases, the most similar project returned by the CBR system was a project developed for a regional bank. In principle, there was no apparent correlation between a big nationwide retailer and a regional bank. When analyzed in more detail, the project in the bank had failed due to characteristics that matched the retailer's current situation particularly, the inexperience of their staff working in information technology and their lack of commitment with the project. Furthermore, neither the retailer nor the bank had ever developed a data mining project before. As the project had already been negotiated and there was no possibility of aborting it, the manager made two decisions. Firstly, he demanded full-time dedication of a member from the retailer's technology team and, secondly, he defined as the first project activity, a quick basic training course for the retailer's team about data mining. Thus, it was possible to reduce the risks of the new project, with the support of the experience from a similar project previously developed.

4.1.3 Learned Lessons Database Load and Application

Aiming at the practical application of this module to actual problems and assessing the benefit of its use, a learned lessons database has been collected at NeuroTech and imported by the environment. Interviews and forms collected experience from 10 data mining experts at several levels of the company, ranging from technical staff working in modeling to chief officers at the board of directors. A wide spectrum of 61 learned lessons was documented in 6 variables, namely: stage of the CRISP-DM, task of stage

of the CRISP-DM, date of learning the lesson, expert who learned the lesson, lesson category and lesson description. These 61 lessons were divided into categories in the following proportions: 35.5% in project risk, 24.2% in best practices, 22.6% in technology and 17.7% distributed in other less frequent categories. Furthermore, the 61 learned lessons were also classified in the following types with their respective proportions: 58.1% of guidelines, 22.6% of problems, 16.1% of problem solutions and 3.2% of general spectrum.

The application of this learned lessons database follows the same Case-Based Reasoning methodology and metrics described in the CBR section above. The only differences are the database used and the objective.

Up to now, the learned lessons module has been used in NeuroTech by the operations' manager, mainly at the beginning of the project, as a complement to the risk estimation module. The data mining specialists are also using the module in two situations: corrective or proactive actions. The corrective situation occurs when a new problem is found, for instance, error in the file importation in SPSS. In this case, after the mistake happens, the specialist consults the Learned Lessons database to identify the best solution to the problem. The proactive situation occurs when a new phase of the project begins, for instance, by signaling the risk of disrupting format in the file importation in SPSS. Another proactive action can be taken after having concluded the pre-processing phase and before beginning the application phase of the algorithm. The specialists query the lessons database aiming at verifying if there is any lesson suggested to avoid the same mistakes in the new phase.

Despite the subjective evaluations, some practical actions have been taken by NeuroTech. For instance, according to the operations' manager, a learned lesson has helped reduce the risk of a new project for developing a fraud detection solution in telecommunications. The lesson informed that the first meeting for solution requirements specification should not be accomplished with the client's business and information technology teams separately, i.e., a learned lesson informed that the first requirements specification meeting should involve both teams at the same time, otherwise, the lack of understanding and alignment would end up increasing the effort and stress for the entire project. In this scenario, the operations' manager postponed the meeting to a controlled occasion where both teams would be together.

5 Conclusions

This paper has presented an environment for the KDD project development process endowed knowledge reuse at a high level. This environment offers three ways of reusing knowledge: 1) project risk assessment and risk explanation based on the metadata database; 2) reuse of project procedures and settings via Case Based Reasoning on the metadata database; and 3) guidelines, recommendations and warnings from the learned lessons database.

Several examples of the knowledge reuse application to real world problems have been presented in this paper, ranging from supporting the decision of whether starting or not a new "risky" data mining project to finding the most appropriate data transformation and parameter settings along its data mining solution development. The experiments have shown that the risk assessment at the beginning of a project along with the risk conditions help developing a high quality project leading to

solutions with high chances of matching the clients' expectations within the schedule and budget planned.

The recent success of NeuroTech on the PAKDD 2007 Competition (First Runner-up) [14] and publication in IJCNN09 [15] has already proved these ideas.

Despite its breadth in terms of managing KDD knowledge, this work has been constrained to the boundaries of binary classification problems. Extensions to other types of problems which were kept out of its scope (e.g. time series forecasting) are already under investigation and will demand a lot of effort.

At the moment, the environment is in full application to real world problems and, soon, there will be enough metadata for presenting results with statistical significance.

References

1. Fayyad, U.M., Piatetsky-Shapiro, G., Smyth, P.: From data mining to knowledge discovery: an overview. In: Advances in knowledge discovery and data mining, pp. 1–34 (1996)
2. Shearer, C.: The CRISP-DM model: the new blueprint for data mining. Journal of Data Warehousing 5, 13–22 (2000)
3. Bartlmae, K.,Riemenschneider, M.: Case based reasoning for knowledge management in kdd projects. In: Proceedings of the 3rd International Conference on Practical Aspects of Knowledge Management. PAKM 2000, Basel Switzerland (2000)
4. Rodrigues, M.d.F., Ramos, C., Henriques, P.R.: How to Make KDD Process More Accessible to Users. In: ICEIS, pp. 209–216 (2000)
5. Brachman, R.J., et al.: Integrated Support for Data Archaeology. International Journal of Intelligent and Cooperative Information Systems 2(2), 159–185 (1993)
6. Wirth, R., et al.: Towards Process-Oriented Tool Support for Knowledge Discovery in Databases. In: Principles of Data Mining and Knowledge Discovery, Trondheim, Norway, pp. 243–253 (1997)
7. Engels, R.: Component-based User Guidance for Knowledge Discovery and Data Mining Processes. Universität Karlsruhe (1999)
8. Bernstein, A., Hill, S., Provost, F.: Intelligent assistance for the data mining process: An ontology-based approach (2002)
9. King, R.D.: The STATLOG Project 2007, Department of Statistics and Modelling Science (2007), `http://www.up.pt/liacc/ML/statlog/index.html`
10. Neaga, I.: Framework for Distributed Knowledge Discovery Systems Embedded in Extended Enterprise, in Manufacturing Engineering, Loughborough, United Kingdom Loughborough University (2003)
11. Kimball, R.: The Data Warehouse Lifecycle Toolkit. John Wiley & Sons, New York (1998)
12. Conover, W.J.: Practical Nonparametric Statistics, vol. 3. John Wiley & Sons, New York (1999)
13. Aamodt, A., Plaza, E.: Case-Base Reasoning: Foundational Issues. Methodological Variations and Systems Approaches AICOM 7(1) (1994)
14. Adeodato, P.J.L., et al.: The Power of Sampling and Stacking for the PAKDD 2007 Cross-Selling Problem. International Journal of Data Warehousing & Mining 4(2), 22–31 (2008)
15. Adeodato, P., et al.: The Role of Temporal Feature Extraction and Bagging of MLP Neural Networks for Solving the WCCI 2008 Ford Classification Challenge. In: International Joint Conference on Neural Networks. IJCNN 2009 (accepted) (2009)

Enhancing Text Clustering Performance Using Semantic Similarity

Walaa K. Gad and Mohamed S. Kamel

Department of Electrical and Computer Engineering, University of Waterloo
Waterloo, Ontario, N2L 3G1, Canada
{walaakh,mkamel}@pami.uwaterloo.ca

Abstract. Text documents clustering can be challenging due to complex linguistics properties of the text documents. Most of clustering techniques are based on traditional bag of words to represent the documents. In such document representation, ambiguity, synonymy and semantic similarities may not be captured using traditional text mining techniques that are based on words and/or phrases frequencies in the text.

In this paper, we propose a semantic similarity based model to capture the semantic of the text. The proposed model in conjunction with lexical ontology solves the synonyms and hypernyms problems. It utilizes WordNet as an ontology and uses the adapted Lesk algorithm to examine and extract the relationships between terms. The proposed model reflects the relationships by the semantic weighs added to the term frequency weight to represent the semantic similarity between terms.

Experiments using the proposed semantic similarity based model in text clustering are conducted. The obtained results show promising performance improvements compared to the traditional vector space model as well as other existing methods that include semantic similarity measures in text clustering.

Keywords: Semantic similarity measures, Adapted Lesk algorithm, Word sense disambiguation, WordNet.

1 Introduction

Text clustering aims at partitioning text documents into related clusters, and discovering the implicit knowledge between clusters. Text clustering has been applied to many applications: indexing, information retrieval, browsing large document collections, management and mining text data on the web. Text clustering can be challenging due to the high dimensionality and complex linguistics properties of text documents.

Most text clustering methods use the traditional term based vector space model (VSM) [1]. VSM represents each term by its frequency. The term may be in the form of a single word and/or a phrase [2,3]. The frequency weight reflects the importance of the term in the document. The term frequency may not discover that two terms are similar because they are lexicographically different. Therefore, clustering based on VSM will fail to group documents that are semantically similar but lexicographically different.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 325–335, 2009.
© Springer-Verlag Berlin Heidelberg 2009

In order to deal with this problem, researchers have focused on finding the semantic features of the document. They integrate ontologies and background knowledge into text clustering process [4,5].

WordNet synsets are used to augment document vector by terms synonyms and achieve better results than the traditional VSM [6]. A similar technique is adapted in document clustering using sense-no and offset of the term in the ontology. The sense-no and offset map the terms to their senses and these senses construct the feature vector to represent the documents [7]. The clustering performance is improved but the statistical analysis showed that this improvement is not significant [7].

In this paper, we propose a new semantic similarity based model (SSBM) and use this model in document text clustering. The model analyzes a document to get the semantic content. The SSBM assigns new weights to reflect the semantic similarities between terms. Higher weights are assigned to terms that are semantically close. In our model, each document is analyzed to extract terms considering stemming and pruning issues. We use the adapted Lesk algorithm to get the semantic relatedness for each pair of terms. SSBM solves the ambiguity and synonym problems that may lead to erroneous grouping and unnoticed similarities between text documents.

Results show that SSBM has a significant improvement of clustering performance over VSM as well as other methods that use semantic similarities [4,5]. The SSBM has a promising performance due to its insensitivity to noisy terms that may lead to incorrect results. We perform the clustering using bisecting kmeans and kmeans algorithms and assess the performance in terms of Fmeasure, Purity and Entropy measures.

The rest of the paper is organized as fellows. Section 2 introduces a brief review of relevant semantic similarities measures. The proposed semantic similarity based model is presented in section 3. Test data, and results are described in section 4. Finally, conclusions are discussed in section 5.

2 Semantic Similarity Measures

The literature has many methods for computing the semantic similarity between terms represented in ontologies. Most semantic similarity measures have been used in conjunction with WordNet. WordNet [8] is an online lexical system developed at Princeton University. WordNet is organized into taxonomic hierarchies. Nouns, verbs, adjectives and adverbs are grouped into synonym sets (synsets). The synsets are related to other synsets higher or lower in the hierarchy by different types of relationships. The most common relationships are the Hyponym/Hypernym (Is-A relationships), and the Meronym/Holonym (Part-Of relationships).

In the following, we review some of relevant semantic similarity measures. Semantic similarity measures can be classified into four categories [9]: Edge Counting Measures, Information Content Measures, Feature Based Measure and Hybrid Methods.

Edge Counting Measures. These measures consider where the two terms t_1 and t_2 are in the taxonomy. Thus, the more similar two terms are, the more links between them and the more closely they are. Shortest Path method [10] measures how the terms close in the taxonomy. Wu and Palmer consider the position of the terms in the taxonomy relatively to the position of the most specific common terms, and define similarity as

a function of a path length that linking the two terms and their positions [11]. Li et al. combine the shortest path, and the depth in the taxonomy of the most specific common term t [12].

Information Content Measures. This method is proposed to overcome the problem of path based method that is based only on taxonomy links. Lord et al. propose a simple method to measure the similarity by using the probability of the most specific shared parent [13]. Resnik signifies that the more information two terms share in common, the more similar they are. The information shared by two terms is indicated by the information content of the term that subsumes both in the taxonomy [14]. Jiang and Conrath propose a combined method which considers both path based method and information content method [15].

Feature based Measure. This method describes the features and the properties of the terms being measured. The features of the terms indicate the terms definitions and their relationships to other similar terms in the taxonomy. The more common features two terms have and the less non common characteristics they have, the more similar the terms are [16].

Hybrid Methods. These methods combine ideas from the above presented approaches, considering the path connecting the terms in the taxonomy, links of the terms with their parents as well as features of the terms.

3 Semantic Similarity Based Model (SSBM)

In the proposed model, a term is defined as a stemmed nonstop word. We have processed text documents using the Porter stemmer [17]. The stemmed terms are used to construct the document vectors. Stemming has only been performed for terms that do not appear in WordNet as lexical entries. The morphological capabilities of WordNet are used to those terms to improve the results. Pruning is applied to eliminate document vectors and prune infrequent terms that may affect the clustering results. Rare terms may add noise and do not help in discovering appropriate clusters.

The term frequency and inverse document frequency $tf.idf$ [1] is used to compute terms weights. The $tf.idf$ of term i in document j is defined by:

$$tf.idf(j,i) = \log(tf(j,i)+1) * \log(\frac{|D|}{df(i)})$$

where $df(i)$ is a document frequency of term i that indicates how many documents term i appears. $tf(j,i)$ how many times term i appears in document j.

$tf.idf$ assigns larger weights to terms that appear relatively rarely through the corpus, but very frequently in individual documents. $tf.idf$ measures 14% improvement in recall and precision in comparison to the standard term frequency tf [1].

SSBM is proposed to group the documents that may not contain the same identical terms. Two documents can be similar and belong to the same category although they have few common terms. VSM treats the terms as an independent features, and no relationships between them. Using VSM, term frequency reflects the term importance

in the document. Thus, VSM will not recognize the semantic meaning of the terms. Our objective is extract semantically similar terms based on WordNet and disambiguates the polysemous terms based on the adapted Lesk algorithm and the document context.

In this model, the term weight is updated and adjusted based on its relationships with other semantically terms that co-occur in the document. The updated semantic weight is called $\tilde{w}_{ji_1}$:

$$\tilde{w}_{ji_1} = w_{ji_1} + \sum_{\substack{i_2=1 \\ i_2 \neq i_1}}^{m} w_{ji_2}.sim_{Lesk}(i_{j1}, i_{j2})$$

where w_{ji_1} is the term frequency of term i_1 in document j, $sim_{Lesk}(i_1, i_2)$ is the semantic information between terms i_1 and i_2 using the adapted Lesk algorithm, and m is the number of terms in document j.

The SSBM updates the documents vectors in VSM to integrate such semantic weights to terms frequency using the adapted Lesk algorithm. The adapted Lesk algorithm solves the problem of finding the correct sense of a word in a context. This algorithm is based mainly on the original Lesk algorithm and extends the glosses comparisons to take the advantage of WordNet relationships.

The Lesk algorithm [18] disambiguates the sense of a polysemous word based on its context, and the definition of all senses in the dictionary. The Lesk algorithm disambiguates words in short phrases. The lexicon definition, gloss, of each sense of a word is compared to the glosses of every other word in the phrase. All the words occurring in the sense definition compose the sense bag. The sense whose gloss shares the largest number of words in common with the glosses of the other words is assigned to the word. The Lesk algorithm relies on glosses found in traditional dictionaries as Oxford advanced Learners [18]. The original Lesk algorithm only considers overlaps among the glosses of the word and those that surround it in the given context. This is a significant limitation because dictionary glosses tend to be short and do not provide sufficient vocabulary to make distinctions in relatedness. The Adapted Lesk algorithm [19] takes the advantage of highly interconnected relations that WordNet offer, and computes the relatedness between terms by comparing the glosses of synsets that are related to terms through relations of WordNet. The relatedness is not only based on glosses overlaps but also between the glosses of the hypernym, hyponym, meronym, holonym and troponym synsets of the input synsets, as well as between synsets related to input terms.

The proposed model works in conjunction with the adapted Lesk algorithm to find the semantic information between each pair of terms, and their relationships with other semantically similar terms within the same document. We use the adapted Lesk in our semantic similarity based model (SSBM) due to:

1- Get the advantage of extended glosses overlaps and relations of WordNet.
2- The adapted Lesk has a good performance compared to other relatedness and similarity measures [7,20].

In our model, we adapt the cosine similarity measure to calculate the cosine of the angle between the two document vectors d_{j_1} and d_{j_2}:

$$\cos(d_{j_1}, d_{j_2}) = \frac{\boldsymbol{d}_{j_1} \cdot \boldsymbol{d}_{j_2}}{||d_{j_1}|| \cdot ||d_{j_2}||} = \frac{\sum_{i=1}^{m} \tilde{w}_{j_1 i} \cdot \tilde{w}_{j_2 i}}{\sqrt{\sum_{i=1}^{m} \tilde{w}_{j_1 i}^2} \sqrt{\sum_{i=1}^{m} \tilde{w}_{j_2 i}^2}}$$

where $\tilde{w}_{j_1 i}$ represents the semantic weight of term i in document d_{j_1} and $\tilde{w}_{j_2 i}$ represents the semantic weight of term i in document d_{j_2} in our proposed model. The similarity measure above has a value between [0,1].

4 Experimental Analysis

4.1 Datasets

he experimental setup consisted of two datasets, Reuters-21578 and 20-Newsgroup text collections. In Reuters-21578, about half of the documents are annotated with category labels, some documents belong to more than one class. Some documents are not assigned to any class. We performed some operations to prepare our base corpus. We selected only the documents that are assigned to one topic. All documents with an empty document body are also discarded. We derived dataset D01 using a process similar to [4]. We restricted the maximum category size to 100 documents and discarded categories with less than 15 documents. Categories that have more than 100 documents are reduced by sampling.

Similar to [5], We extracted four datasets with the same configurations from 20-Newsgroups. Dataset D02 and D04 contain categories with very different topics (comp.graphics, rec.sport.baseball, sci.space, talk.politics.mideast), while D03 and D05 consist of categories in similar topics (comp.graphics, comp.os.ms-windows, rec.autos, sci.electro-nics). Datasets D02 and D03 are of equal size categories (balanced), while datasets D04 and D05 are not of equal size categories (unbalanced). Table 1 summaries all the datasets.

Table 1. Summary of Datasets

Datasets	Size	k	Categories
D01	2619	60	Unbalanced/Different
D02	400	4	Balanced/Different
D03	400	4	Balanced/Similar
D04	299	4	Unbalanced/Different
D05	299	4	Unbalanced/Similar

4.2 Evaluation Measures

We evaluated the effectiveness of SSBM using three clustering quality measures, Fmeasure, Purity and Entropy [21]. Fmeasure combines Precision and Recall measures. Precision measure is the percentage of relevant documents retrieved with respect to the

number of retrieved documents. Recall measure is the percentage of relevant documents retrieved with respect to the total number of relevant documents in the dataset.

The precision and recall of a cluster c $\in C$ for a given class $\ell \in L$ are given respectively by:

$$P(c,\ell) = \frac{|c \cap \ell|}{|c|}, \mathrm{R}(c,\ell) = \frac{|c \cap \ell|}{|\ell|}$$

$$Fmeasure(c,\ell) = \frac{2\mathrm{PR}}{\mathrm{P}+R}$$

where $|c \cap \ell|$ is the number of documents belonging to cluster c and class ℓ, $|c|$ is the size of cluster c, $|\ell|$ is the size of class ℓ.

The second measure is the Purity. The overall value for Purity is computed by taking the weighted average of maximal precision values:

$$Purity(C,L) = \sum\nolimits_{c \in C} \frac{|c|}{|D|} \max_{\ell \in L} P(c,\ell)$$

The third measure is The Entropy, which measures how homogeneous a cluster is. The higher homogeneity of a cluster, the lower Entropy is, and vice versa. Entropy of cluster c is $E(c) = \sum_{\ell \in L} \mathrm{P}(c,\ell).\log(c,\ell)$ and the entropy of all the clusters is the sum of the entropy of each cluster weighted by its size.

$$E(C) = \sum_{c \in C} \frac{|c|}{|D|} P(c)$$

4.3 Results and Analysis

Bisecting kmeans and kmeans techniques are chosen for testing the effect of the semantic similarity based model on text document clustering. Each evaluation result is an average of 20 runs to alleviate the effect of a random factor. The Fmeasure, Purity and Entropy values are the average over the 20 runs. Our objective is to maximize the Fmeasure, Purity and minimize the Entropy. We compared the results of our semantic similarity based model (SSBM) to the vector space model (VSM) as a baseline.

Both VSM and SSBM have the same preprocessing techniques, stopwords removal, stemming and pruning. Figures 1, 2 and 3 show the comparisons of bisecting kmeans clustering quality based on VSM and SSBM, while Figures 4, 5 and 6 show the comparisons of kmeans clustering quality based on VSM and SSBM,

The right bar shows the performance including the semantic similarity based model, and the left bar shows the performance of traditional VSM based on the term frequency.

The experimental results show that the proposed SSBM improves the clustering quality better than the traditional term VSM for all datasets. In addition, we compare SSBM performance to other methods that integrate semantic similarities into text clustering. The semantic similarity based model outperforms the methods introduced in [6] and [5]. These methods are background and ontology methods. The background knowledge is evaluated on D01 dataset compared to the VSM as a baseline using bisecting

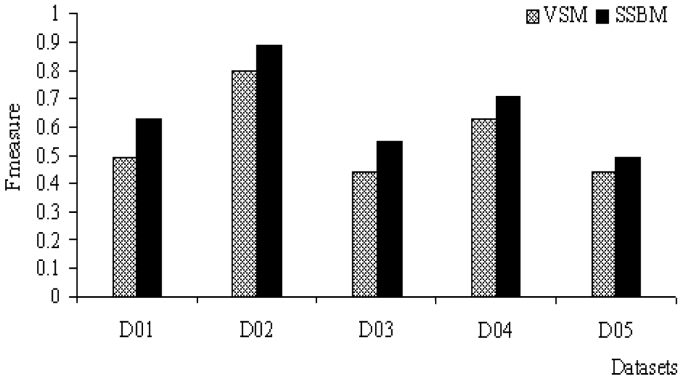


Fig. 1. Comparison of bisecting kmeans clustering quality based on VSM and SSBM in terms of Fmeasure

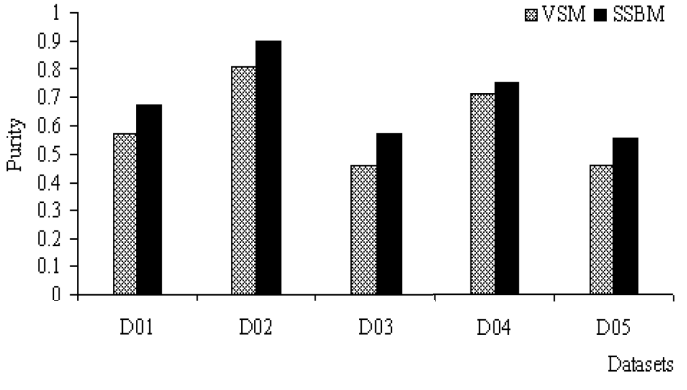


Fig. 2. Comparison of bisecting kmeans clustering quality based on VSM and SSBM in terms of Purity

kmeans in terms of the Purity quality measure. Ontology method tests the performance on four datasets D02 to D05. Feature weighted kmeans (FW-kmeans) and kmeans are implemented to show the improvement in clustering in terms of Fmeasure and Entropy quality measures.

Table 2 shows the relative improvement of the three methods: Background knowledge, Ontologies and our proposed semantic similarity based model SSBM. All methods use the same dataset configurations and compared to the VSM as a baseline. Relative improvement of Fmeasure, Purity and Entropy are denoted by RFm, REn and RPu respectively.

Our experiments show that the improvements range from 10% to 26% increase in Fmeasure, and 18% to 31% drop in Entropy compared to method introduced in [6]. The SSBM outperforms the background knowledge method by 18% in Purity clustering quality measure.

The SSBM has a better performance than VSM due to the contribution of non identical but semantically similar terms. In other semantic similarity methods, terms

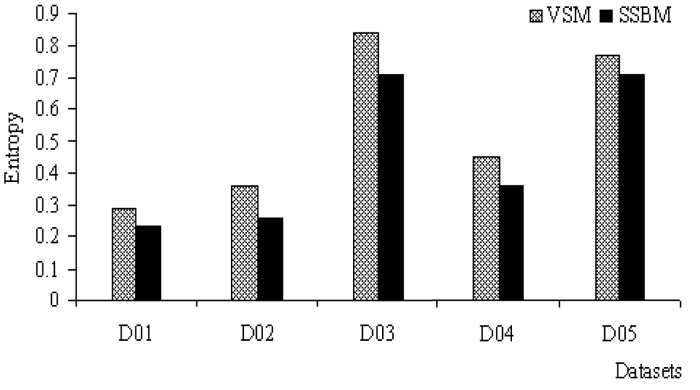


Fig. 3. Comparison of bisecting kmeans clustering quality based on VSM and SSBM in terms of Entropy

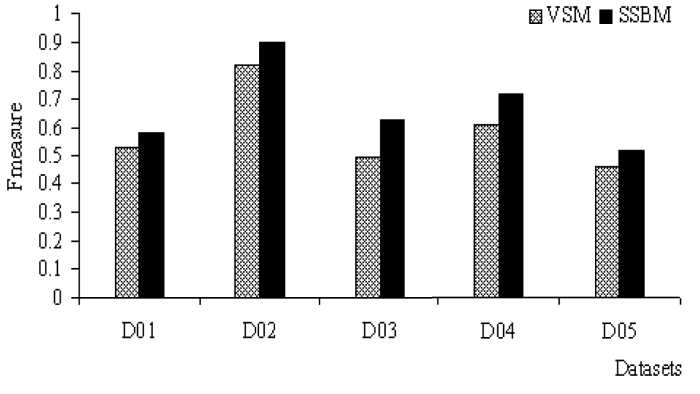


Fig. 4. Comparison of kmeans clustering quality based on VSM and SSBM in terms of Fmeasure

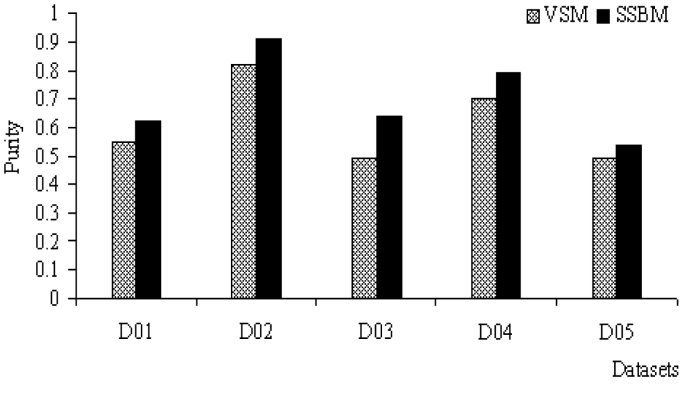


Fig. 5. Comparison of kmeans clustering quality based on VSM and SSBM in terms of Purity

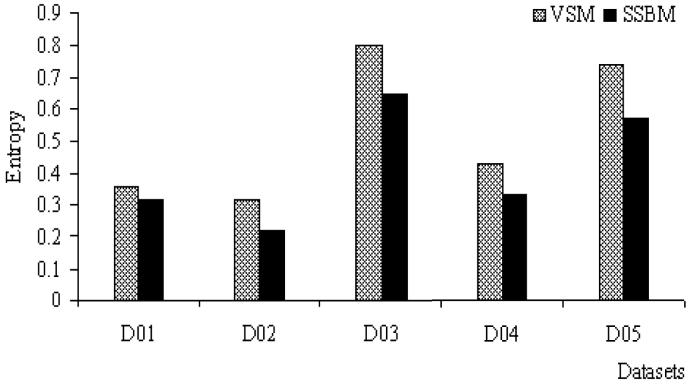


Fig. 6. Comparison of kmeans clustering quality based on VSM and SSBM in terms of Entropy

Table 2. Relative improvements of Background knowledge, Ontologies and SSBM

Datasets		Background		SSBM
		Bisecting kmeans		
D01	RIPu	8.4%		17.54%
		Ontologies		SSBM
		kmeans	FW-kmeans	kmeans
D02	RIFm	4.80%	4.38%	9.75%
	RIEn	5.71%	16.02%	31.25%
D03	RIFm	7.35%	6.18%	26%
	RIEn	9.89%	17.24%	18.75%
D04	RIFm	0.88%	0.69%	18.03%
	RIEn	4.10%	4.75%	23.25%
D05	RIFm	4.91%	4.61%	13.04%
	RIEn	13.12%	18.37%	23%

synonyms are added to document vectors. WordNet provides up to five senses for a term. It means that for a one correct sense there are many incorrect senses added. This may lead to extra overlap between documents due to noisy senses. The proposed model reweights and assigns new semantic weights to the terms using the adapted Lesk algorithm. The adapted Lesk algorithm disambiguates the senses of a polysemous word based on its context in the document. Therefore, the reason behind this improvement is that SSBM captures the importance of the terms that are related to the document topic, and is insensitive to noise when calculating document vector similarities.

5 Conclusions

In this paper, the semantic similarity based model (SSBM) is proposed. The proposed model represents the text document by exploiting the terms semantics. The SSBM introduces the WordNet ontology into text clustering using the adapted Lesk algorithm to assign new weights to document terms.

The new weight reflects the semantic relatedness between co-augmented terms. Higher weights are assigned to terms that reflect the document meaning. The SSBM is less sensitive to noise due to the new term weight and semantic similarity calculations based on terms context. The proposed model solves the ambiguity and synonymy problems in conjunction with the Adapted Lesk algorithm, and captures the semantic similarities of documents. The evaluation demonstrates very promising performance compared to traditional term based vector space model (VSM) and other semantic methods that integrate semantics into text clustering. We implement kmeans and bisecting kmeans to assess the performance of text clustering in terms of Fmeasure, Purity and Entropy clustering measures.

References

1. Salton, G., McGill, M.: Introduction to Modern Information Retrieval. McGraw-Hill, New York (1983)
2. Hammouda, K., Kamel, M.: Efficient Phrase-based Document Indexing for Web Document Clustering. IEEE Transactions on Knowledge and Data Engineering 16, 1279–1296 (2004)
3. Shehata, S., Karray, F., Kamel, M.: A Concept-Based Model for Enhancing Text Categorization. In: The 13th ACM SIGKDD international Conference on Knowledge Discovery and Data Mining, pp. 629–637 (2007)
4. Hotho, A., Staab, S., Stumme, G.: WordNet Improve Text Document Clustering. In: SIGIR 2003 Semantic Web Workshop, pp. 541–544 (2003)
5. Jing, L., Zhou, L., Ng, M., Huang, Z.: Ontology-based Distance Measure for Text Clustering. In: IAM SDM Workshop on Text Mining (2003)
6. Sedding, J., Dimitar, K.: WordNet-based Text Document Clustering. In: COLING 2004 3rd Workshop on Robust Methods in Analysis of Natural Language Data, pp. 104–113 (2004)
7. Wang, Y., Hodges, J.: Document Clustering with Semantic Analysis. In: The 39th Annual Hawaii International Conference on System Sciences, 2006. HICSS 2006, vol. 3, p. 54c (2006)
8. Fellbaum, C.: WordNet: An Electronic Lexical Database. MIT Press, Cambridge (1998)
9. Budanitsky, A., Hirst, G.: Evaluating WordNet-based Measures of Lexical Semantic Relatedness. Computational Linguistics 32, 13–47 (2006)
10. Rada, R., Mili, H., Bickell, E., Blettner, B.: Development and Application of a Metric on Semantic Nets. IEEE Transactions on Systems, Man and Cybernetics 19, 17–30 (1989)
11. Wu, Z., Palmer, M.: Verb Semantics and Lexical Selection. In: The 32nd Annual Meeting of the Association for Computational Linguistics, pp. 133–138 (1994)
12. Li, Y., Zuhair, A., McLean, D.: An Approach for Measuring Semantic Similarity between Words Using Multiple Information Sources. IEEE Transactions on Knowledge and Data Engineering 15, 871–882 (2003)
13. Lord, P., Stevens, R., Brass, A., Goble, C.: Semantic Similarity Measures as Tools for Exploring the Gene Ontology. In: The 8th Pacific Symposium on Biocomputing, vol. 8, pp. 601–612 (1997)
14. Resnik, P.: Using Information Content to Evaluate Semantic Similarity in Taxonomy. In: The 14th international joint conference Artificial Intelligence, pp. 448–453 (1995)
15. Jiang, J., Conrath, D.: Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy. In: International Conference on Research in Computational Linguistics, pp. 19–33 (1997)
16. Tversky, A.: Features of Similarity. Psychological Review 84, 327–352 (1977)
17. Porter, M.: An algorithm for Suffix Stripping. Program 14, 130–137 (1980)

18. Lesk, M.: Automatic Sense Disambiguation Using Machine Readable Dictionaries: How to Tell a Pine Cone from an Ice Cream Cone. In: The ACM SIG-DOC Conference, pp. 24–26 (1986)
19. Banerjee, S., Pedersen, T.: Extended Gloss Overlaps as a Measure of Semantic Relatedness. In: 8th International Joint Conference on Artificial Intelligence (IJCAI 2003), pp. 805–810 (2003)
20. Patwardhan, S., Banerjee, S., Pedersen, T.: Using Measures of Semantic Relatedness for Word Sense Disambiguation. In: Gelbukh, A. (ed.) CICLING 2003. LNCS, vol. 2588, pp. 241–257. Springer, Heidelberg (2003)
21. Steinbach, M., Karypis, G., Kumar, V.: A Comparison of Document Clustering Techniques. In: KDD Workshop on Text Mining (2000)

Stereo Matching Using Synchronous Hopfield Neural Network

Te-Hsiu Sun

Department of Industrial Engineering and Management
Chaoyang University of Technology, ROC, Taiwan
thsun@mail.cyut.edu.tw

Abstract. Deriving depth information has been an important issue in computer vision. In this area, stereo vision is an important technique for 3D information acquisition. This paper presents a scnaline-based stereo matching technique using synchronous Hopfield neural networks (SHNN). Feature points are extracted and selected using the Sobel operator and a user-defined threshold for a pair of scanned images. Then, the scanline-based stereo matching problem is formulated as an optimization task where an energy function, including dissimilarity, continuity, disparity and uniqueness mapping properties, is minimized. Finally, the incorrect matches are eliminated by applying a false target removing rule. The proposed method is verified with an experiment using several commonly used stereo images. The experimental results show that the proposed method solves effectively the stereo matching problem and is applicable to various areas.

Keywords: Stereo matching, Correspondence problem, Synchronous Hopfield neural network, Computer vision.

1 Introduction

Measuring depth information is an important issue in many areas such as metrology and quality control. Among depth measuring techniques, stereo vision is an economic and passive method to derive effective 3D information of objects. The depth information perceived in stereovision systems depends fully on the disparity between the two images. A stereovision system usually employs a pair of cameras that simultaneously take images of an object from different angles. The key of the depth information is to determine which point in one image corresponds to a given point in the other image (called the "correspondence problem" or "stereo matching problem"). The corresponding points in the stereo images are called the conjugate points. Many different approaches have been developed to solve this difficult problem. In general, these approaches can be classified into two main categories: feature-based matching (FBM), and area-based matching (ABM) [4]. For the ABM methods, each point of an image is matched as the center of a small window of pixels in a reference image,

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 336–347, 2009.
© Springer-Verlag Berlin Heidelberg 2009

where either a difference or similarity measure is minimized or maximized. The common approaches in this category include cross-correlation [29], least square [1], [5], [8], and [9], and absolute deviation matching methods [2]. For the FBM methods, features such as boundaries, edges, corner points, lines, arcs, and regions are extracted and then used as a matching set to facilitate the matching process instead of an area [7], [17], [21], and [27]. When using points as matching features, the selection of feasible matching candidates from the enormous features becomes the first issue. In general, the FBM method derives a sparse map and leaves the rest of the area to be interpolated later.

Hopfield and Tank [11]-[13] developed Hopfield neural networks, which have been widely applied to the problems such as weighted point matching [10], path determination [3], analogy-to-digital (A/D) conversion [30], and pattern recognition [24]. Mousavi and Schalkoff [21] used Hopfield neural networks to minimize an energy function subject to the similarity and epipolar constraints. Later, they improved the formulation by using the intra-scanline and inter-scanline constraints to enforce the energy function [22]. Parvin and Medioni [28] used a Hopfield neural network for a multi-scale strategy in matching the extracted features for the correspondence of 3D objects. In their network, local, adjacency and global constraints were satisfied. Wolfe also solved the correspondence problem and the pose estimation problem by Hopfield neural networks [31] and [32]. They used Hopfield networks to assign the mapping relationship that scores the highest. Hu and Siy [14] proposed a novel class of optimization problems called the "Picking Stone Problem" (PSP), which was solved under uniqueness and ordering constraints by Hopfield neural network. Nasrabadi and Li [24] developed a two-dimensional binary Hopfield neural network for object recognition, and subsequently they extended their research to the stereo features matching (Nasrabadi and Choo, 1992). Lee *et al.* [18] applied a Hopfield neural network to the stereo correspondence for extracting the 3D structure of a scene. Similarity, smoothness and uniqueness constraints were transformed into the form of an energy function. Ruichek and Postaire [27] built an energy function mapped onto a two-dimensional Hopfield neural network for minimization. Pajares *et al*. [26] presented a relaxation approach using Hopfield neural networks to solve the global stereo matching problem. They used edges as the matching primitives and the similarity, smoothness, and uniqueness properties are adopted for matching. Tien and Chang [16] used Hopfield neural networks to solve the stereo matching problem, and then applied the 3D information for inspection. Mortara and Spagnuolo [19] presented a solution to the correspondence problem for polygon blending by matching the skeleton that was approximated by morphological characterization of the shape. Giaguinto *et al.* [6] used a cellular neural network (CNN), which minimized the energy function for a real-time stereo matching problem. In general, the drawbacks of these methods are numerous incorrect matches produced and heavy computation and memory space required.

This study describe a new scanlined based stereo matching method that first represents the stereo matching problem as an energy function and minimized it by a

designed synchronous Hopfield neural network (SHNN). Then, a statistical false target removing rule is proposed to screen out the incorrect matches, and the disparity map is obtained as the final result.

2 Scanline-Based Stereo Matching Problem

Stereo matching problem is a complicated problem with high computation complexity because of the enormous number of mapping features and the ambiguity between two sets of corresponding feature points. In order to reduce the complexity of the problem, feature points are selected and matched along the epipolar lines. In addition, domain, dissimilarity, disparity, continuity, and uniqueness properties in stereo matching system are taken to restrict the matching candidates in a certain area, and thus this problem can be solved more effectively. Given *m* features in the left image and *n* features in the right image of a scanline as shown in Fig. 1, the matching properties are discussed below.

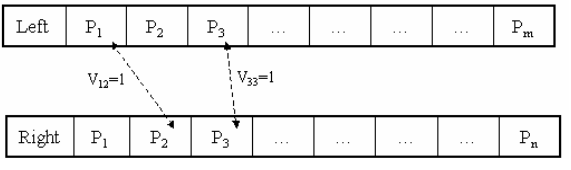


Fig. 1. Scanline matching with the left and right arrays

(i) Domain property. V_{ij} represents the matching relationship between the left i^{th} feature and the right j^{th} feature. This indicator variable takes values on $\{0, 1\}$ to denote an active match. For example, V_{12}=1 represents that the first left feature point matches the second right feature points as shown in Figure 1.
(ii) Dissimilarity property. The dissimilarity property describes the difference between two matching feature points. When two extracted feature points are not conjugate, the difference of their attributes in the left and right images should be large, and vice versa. In order to measure the dissimilarity, the invariant factors such as the mean and variance of intensity in a sized window *w* are commonly used. The mean μ, and the variance σ^2, of the intensity of a window are defined as

$$\mu_i(x,y) = \sum_{k=x-\lfloor w/2 \rfloor}^{x+\lfloor w/2 \rfloor} \sum_{l=y-\lfloor w/2 \rfloor}^{y+\lfloor w/2 \rfloor} \frac{I(k,l)}{w^2},$$

$$\sigma_i^2 = \frac{\sum_{k=x-\lfloor w/2 \rfloor}^{x+\lfloor w/2 \rfloor} \sum_{l=y-\lfloor w/2 \rfloor}^{y+\lfloor w/2 \rfloor} [I(k,l) - \mu_i(x,y)]^2}{9} \quad (1)$$

where $I(x, y)$ is the intensity (gray level) at the feature point (x, y) and $i \in \{r, l\}$ indicates the left or right image. Two other effective invariant factors are the gradient magnitude and direction of feature points, which are defined as

$$mag_i(\nabla f) = (G_x^2 + G_y^2)^{1/2}, \tag{2}$$

$$\alpha_i(x, y) = \tan^{-1}\left(\frac{G_y}{G_x}\right), \tag{3}$$

where G_x and G_y are the gradient in the x direction and y direction (Wolfe and Magee, 1990) and $i \in \{r, l\}$ indicates the left or right image. Therefore, the measure of the dissimilarity property for the i^{th} and j^{th} feature points on the left and right scanlines is defined as

$$S_{ij}=t_1(\mu_l-\mu_r)^2+t_2(\sigma_l-\sigma_r)^2+t_3(mag-mag)^2+t_4(\alpha_l-\alpha_r)^2,$$

$$1\le i\le m,\ 1\le j\le n \tag{4}$$

where the coefficients, t_1, t_2, t_3 and t_4 are the weights for the squared difference components. The energy function of dissimilarity is derived as

$$E_s=\sum_{i=1}^{m}\sum_{j=1}^{n} S_{ij}V_{ij}\ . \tag{5}$$

(iii) Disparity property. The range of disparity is determined by the geometry of objects. This property explains that the disparities of mapping features should be limited in a certain range ($d_{\min}$, $d_{\max}$). Accordingly, the property is described as

$$d_{\min} \le d_{ij}V_{ij} \le d_{\max} \tag{6}$$

$$d_{ij}=|x_j - x_i| \quad 1\le i\le m,\ 1\le j\le n,$$

where x_i and x_j are the coordinates of the i^{th} and j^{th} feature points on the left and right scanlines and $d_{\max}$ is determined empirically. By setting $d_{\min}$=1, the energy function of disparity is expressed as

$$E_d=\sum_{i=1}^{m}\sum_{j=1}^{n}[d_{ij}V_{ij}-d_{\max}]^2\ . \tag{7}$$

(iv) Continuity property. Disparities of adjacent feature points should be continuous in a stereovision system. This property describes the following relationship between feature points in a scanline as

$$|d_{ij}-d_{i+1,k}|V_{ij}\cdot V_{i+1,k}<\eta\ , \tag{8}$$

$$1\le i\le m,\ 1\le j\le n,\ 1\le k\le n$$

where η is a small number determined empirically. The energy function of continuity is given as

$$\boldsymbol{E}_{\mathrm{c}}=\sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{k=1}^{n}[(d_{ij}-d_{i+1,k})V_{ij}\cdot V_{i+1,k}-\eta]^2\cdot \tag{9}$$

(v) Uniqueness property. The uniqueness property defines the one-to-one relationship between the conjugate feature points as

$$\sum_{i=1}^{m} X_{ij}=1,\ \ 1\le j\le n\ , \tag{10}$$

$$\sum_{j=1}^{n} X_{ij}=1,\ \ 1\le i\le m. \tag{11}$$

The energy functions of uniqueness is represented as

$$\boldsymbol{E}_{\mathrm{u}}=\sum_{i=1}^{m}(1-\sum_{j=1}^{n}V_{ij})^2+\sum_{j=1}^{n}(1-\sum_{i=1}^{m}V_{ij})^2\cdot \tag{12}$$

Therefore, the energy function of the scanline-based stereo matching problem is represented as follows:

$$\boldsymbol{E}=a_1\boldsymbol{E}_{\mathrm{s}}+a_2\boldsymbol{E}_d+a_3\boldsymbol{E}_c+a_4\boldsymbol{E}_u \tag{13}$$

where the empirically determined a_1, a_2, a_3, *and* a_4 are positive coefficients of each energy factors. This energy function is minimized by the proposed SHNN described in the following sections.

3 Proposed Method

The proposed method is designed in three stages, *feature extraction and selection, stereo matching, and false target removing* as described in the following sections.

3.1 Feature Extraction and Selection

The purpose of feature extraction is to obtain the invariant feature points, such as corners and edge points. To find these salient points, many techniques have been developed in the literature [15] and [20]. This study adopts Sobel operator to extract the feature point and then select features with a pre-determined threshold T to for matching. Take the corridor images for example. The image shown in Fig. 2(b) was derived after applying the Sobel operator. By using different thresholds, we derived different numbers of feature points as shown in Fig. 2(c)-(f).



(a) Original left corridor image (b) After Sobel operator (c) T= 160



(d) T=120 (e) T=80 (f) T=40

Fig. 2. Feature extraction with different thresholds

3.2 Stereo Matching Using Synchronous Hopfield Neural Networks (SHNN)

The Hopfield neural network is a single layer feedback network. When the Hopfield neural network is employed to solve the scanline-based stereo problem, V_{ij} is used to denote a neuron representing the matching relationship defined in the domain property. A stochastic activation rule is required when the Hopfield neural network searches for the minimum of the energy function. According to the postulates of Hopfield neural network, it consists of *mxn* neurons with threshold T_{ij}. Let ξ_{klij} denote the weight value connecting the output of the ij^{th} neuron with the input of the kl^{th} neuron. The feedback input to the ij^{th} neuron is equal to the weighted sum as

$$net_{ij} = \sum_{k=1}^{m}\sum_{l=1}^{n} \xi_{ijkl} \cdot V_{ij} - T_{ij} \tag{14}$$

$$1 \le i \le m,\ 1 \le j \le n,\ 1 \le k \le n.$$

The derivation of the activation function Ψ_{ij} and its thresholds T_{ij} are determined by calculating the difference of the energy function between the current state and the previous state

$$net_{ij} = \Psi_{ij} - T_{ij} \tag{15}$$

$$\Psi_{ij} = a_3 \sum_{k=1}^{n} [(d_{ij} - d_{i+1,k}) \cdot V_{i+1,k} + 2\eta] \cdot (d_{ij} - d_{i+1,k}) \cdot V_{i+1,k} \tag{16}$$

$$T_{ij} = a_1 \cdot S_{ij} + a_2 (d_{ij} - 2d_{\max}) d_{ij} + 2a_4 . \tag{17}$$

Then, the change in energy is calculated as

$$\Delta E_{ij} = net_{ij} \times \Delta V_{ij} \tag{18}$$

where $\Delta V_{ij} = V_{ij}^{new} - V_{ij}^{old}$. Contrast to asynchronous Hopfield neural networks, synchronous Hopfield neural networks update a cycle of two states rather than a single state [33]. After a cycle of updating two states, the neuron V_{ij} is updated only when ΔE is less than zero. The state of neurons is continuously updated and fed back to the network. After a certain number of cycles, the network will eventually reach a steady state and an approximate optimal solution will be achieved.

3.3 False Target Removing

After stereo matching, incorrect matches, so called false targets, may be created during the matching process. To remove those incorrect matches, a false target removing process is usually adopted [23]. Generally, epipolar and disparity constraints are the two most commonly used methods. To remove the incorrect matches, a statistical rule is developed to screen out the matches as

$$| x_i - x_j | > \bar{d} + k\delta_d, \qquad \text{for } V_{ij}=1 \tag{19}$$

where

$$\bar{d} = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} | x_i - x_j | V_{ij}}{\sum_{i=1}^{m}\sum_{j=1}^{n} V_{ij}}$$

$$\delta_d = \sqrt{\frac{\sum_{i=1}^{m}\sum_{j=1}^{n} (| x_i - x_j | - \bar{d})^2}{m \times n}},$$

where x_i and x_j are the coordinates of the i^{th} and j^{th} feature points on the left and right scanlines, $\bar{d}$ is the average of derived disparities, and δ_d is the standard deviation of the horizontal disparity after stereo matching, and k is a constant determining the number of standard deviation of disparity. In particular, a small k provides a stronger condition that removes more incorrect matches and derives a smooth disparity map.

4 Implementation

The proposed synchronous Hopfield neural network (SHNN) was implemented in C++ language and run on a Personal computer (ASUS Laptop) with Pentium-III 1133 MHZ CPU and 260MB. To express the depth information explicitly, the disparities of feature points were shown in intensity and rescaled to the range [50, 255]. Experiment I was conducted to determine the coefficients t_i and a_i, and Experiment II was conducted to determine the different combinations of k and d_{max}. In order to

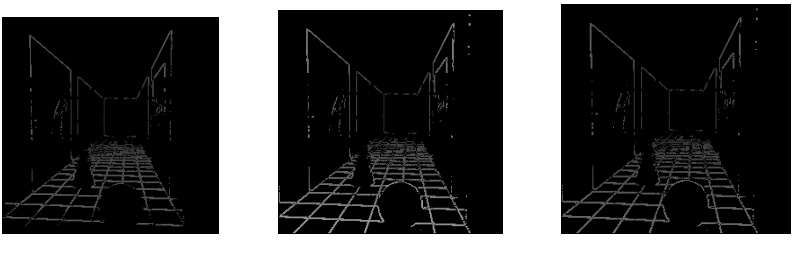
demonstrate the feasibility and applicability of the proposed method, two stereo images were used for verification

4.1 Experiment I – Determination of t_i and a_i

The parameters of the proposed method such as t_i and a_i were determined empirically. In Equation (4), t_1, t_2, t_3 and t_4 are used to determine the impacts of invariant factors including the mean and standard deviation of intensity, gradient magnitude and direction in a sized window. When fixing other parameters, the experimental results showed that fewer incorrect pairs were derived when increasing t_1 and t_2, and more incorrect pairs were derived when increasing t_3 and t_4. Therefore, t_1, t_2, t_3, and t_4 were determined at the ratio of 20: 20: 1: 1. In Equation (13), a_i's are the weights of the energy factors, $\boldsymbol{E}_i$'s. When other parameters remained the same, the experimental results showed that a_4 did not have particular impact on stereo matching under the synchronous Hopfield neural network. After trial-and-error, the parameters a_1, a_2, a_3, and a_4 were determined at the ratio of 1: 100: 100: 1.

4.2 Experiment II – Determination of d_{max} and k

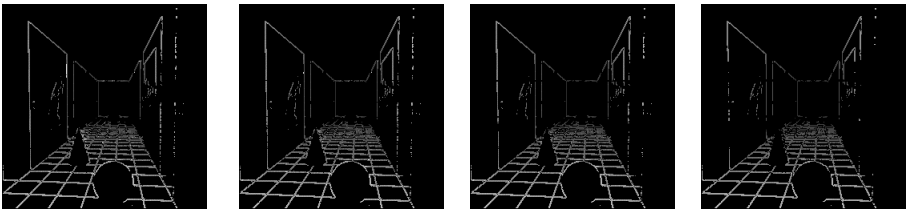
For a given object, its geometry and the distance to the camera determine the disparity in a stereovision system. In Equation (6), $d_{\max}$ is defined as the maximum disparity in the disparity map. Adding this constraint with a small $d_{\max}$ screened out the incorrect matches as well as increasing the risk of losing valuable information. As shown in Fig. 3(a)-(c), the proposed SHNN matched the corridor images when $d_{\max}$= 6, 15 and 25 and consumed 229.901, 230.211, and 236.66 seconds respectively. Using $d_{\max}$= 15, the proposed method derived an excellent disparity map which preserves much of the information, while valuable information was lost when using $d_{\max}$= 6. When $d_{\max}$= 25, the proposed method consumed more computation time without gaining better results than when $d_{\max}$= 15.



(a) $d_{\max}$ =6 (b) $d_{\max}$ =15 (c) $d_{\max}$ =25

Fig. 3. The effects of different $d_{\max}$'s

After matching with SHNN, the derived matched pairs were removed when their disparities were larger than $\bar{d} + k\delta_h$ as in Equation (19). Using a small k may resulted in removing some correct information, while using a large k may retained many false targets. As shown in Fig. 4, when k= 1.5, many false targets were removed without losing valuable information.

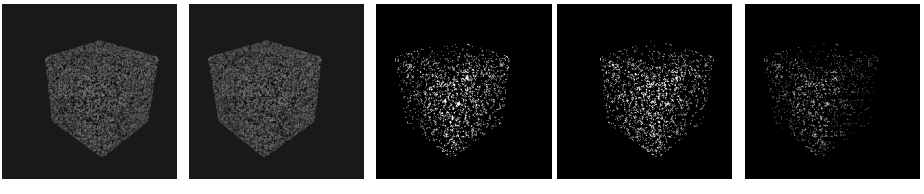


(a) k=2 (b) k=1.5 (c) k=1 (d) k=0.5

Fig. 4. SHNN matching with different k's

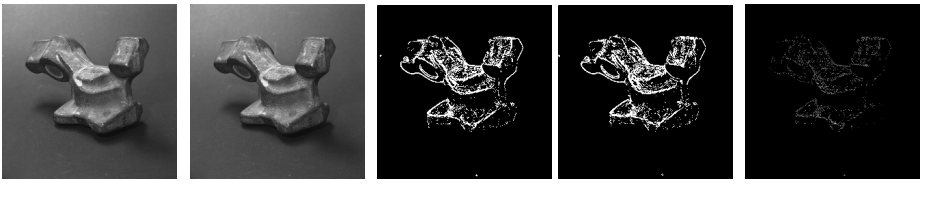
4.3 Verification and Benchmarking

Two stereo images were used to verify the proposed SHNN with the capability and feasibility of stereo matching. Among the testing stereo images as shown in Fig. 5-6 (a) and (b), the cubic images were synthetic and the part images were real. According to previous experiments, the parameters used for the following testing images are listed in Table 1. Features were extracted using the Sobel operator and then thresholded into binary images as shown in Fig. 5-6 (c) and (d). The disparity maps of extracted feature points derived using SHNN and the false target removing were as shown in Fig. 5-6 (e). In addition, the proposed method obtained successfully the depth information for each testing images in acceptable time periods.



(a) Left image (b) Right image (c) Extracted left image (d) Extracted right image (e) Disparity map

Fig. 5. Cubic stereo images and disparity map



(a) Left image (b) Right image (c) Extracted left image (d) Extracted right image (e) Disparity map

Fig. 6. Part stereo images and disparity map

Table 1. Parameters used for testing images

Images	Threshold	No. of STD (k)	d	t_i	a_i
Cubic	110	1.5	60	20:20:1:1	1:100:100:1
Part	40	1.2	20	20:20:1:1	1:100:100:1

5 Conclusions

This study presents a heuristic approach to solve the scanline-based stereo matching problem using the synchronous Hopfield neural network (SHNN. Feature points were extracted and selected using the Sobel operator and a user-defined threshold. Then, the scanline-based stereo matching problem was formulated as an optimization task where an energy function, including dissimilarity, continuity, disparity and uniqueness mapping properties, was minimized. Finally, the incorrect matches were eliminated by applying the proposed false target removing rule. Several experiments were conducted to verify the proposed method. The experimental results showed that the proposed method was feasible and capable of solving the stereo matching problem effectively. The advantage of using scanline-based matching can be explained by the analysis of computation complexity. Theoretically, the smoothness, order, and compatibility constraints usually demand at least $\boldsymbol{O}(m^2 \times n^2)$ complexity of computation, while the proposed method requires $\boldsymbol{O}(m \times n \times k)$. In addition, the proposed method requires $\boldsymbol{O}(m \times n)$ memory space for storing the matching pairs, which is superior to the former [23]. This makes the proposed method suitable for implementing as a PC-based algorithm. In comparison with global matching techniques [16] and [23], their methods require the complexity of $\boldsymbol{O}(M^2 \times N^2)$ or $\boldsymbol{O}(M \times N)$ for computation and $\boldsymbol{O}(M \times N)$ for memory, where M and N are the total numbers of feature points extracted from both images. M ($>>m$) and N ($>>n$) are often large numbers. It is the reason that most of global matching techniques usually run on workstation level computers. Furthermore, a false target removing algorithm was designed to detect mismatches so that the proposed method derives better depth information.

References

1. Ackermann, F.: Digital image correlation-performance and potential application in photogrammetry. Photogrammetric Record 11(64), 429–439 (1984)
2. Calitz, M.-F., Ruether, H.: Least absolute deviation (LAD) image matching. ISPRS Journal of Photogrammetry and Remote Sensing 52, 160–168 (1985)
3. Cavalieri, S., Di Stefano, A., Mirabella, O.: Optimal path determination in a Graph by Hopfield Neural Network. Neural Networks 7(2), 387–404 (1994)
4. Do, K.-H., Kim, Y.-S., Uam, T.-U., Ha, Y.-H.: Iterative relaxational stereo matching based on adaptive support between disparities. Pattern Recognition 31(8), 1049–1059 (1998)
5. Foerstner, W.: On the geometric precision of digital correlation. International Archives of Photogrammetry 24, 176–189 (1982)

6. Giaquinto, N., Savino, M., Taraglio, S.: A CNN-based passive optical range finder for real-time robotic applications. IEEE Transactions on Instrumentation and Measurement 51(2), 314–319 (2002)
7. Grimson, W.: Computational experiments with a feature based stereo algorithm. IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI 7(1), 17–34 (1985)
8. Gruen, A.: Adaptive least square correlation: a powerful image matching technique. ISPRS Journal of Photogrammetry, Remote Sensing and Cartography 14(3), 175–187 (1985)
9. Gruen, A., Agouris, P.: Linear extraction by least squares template matching constrained by internal forces. In: Proceedings of ISPRS Commission III Symposium on Spatial Information from Digital Photogrammetry and Computer Vision, September, vol. 509(30), pp. 316–232 (1994)
10. Hertz, J., Krogh, A., Palmer, R.G.: Weighted point matching, Introduction to the theory of neural computation, pp. 72–79. Addison-Wesley, Reading (1991)
11. Hopfield, J.: Neural networks and physical systems with emergent collective computational abilities. Proc. Nat. Acad. Sci. 79, 2554–2558 (1982)
12. Hopfield, J.: Neurons with graded response have collective computational properties like those of two-state neurons. Proc. Nat. Acad. Sci. 81, 3088–3092 (1982)
13. Hopfield, J., Tank, D.W.: 'Neural' computation of decisions in optimization problems. Biol. Cybern. 52, 141–152 (1985)
14. Hu, J.-E., Siy, P.: An ordering-oriented Hopfield network and its application in stereo vision. In: SPIE 1965, Applications of Artificial Neural Networks, vol. IV, pp. 556–567 (1993)
15. Huang, M.S., Lew, T.S., Wong, K.: Learning and feature selection in stereo matching. IEEE Transactions on Pattern Analysis and Machine Intelligence 16(9), 869–881 (1994)
16. Tien, F.-C., Chang, C.A.: Neural Network for Precise 3D Measurement in Stereo Visio System. International Journal of Production Research 37(9), 1935–1948 (1999) (SCI, IF:0.481)
17. Lee, S.-H., Leou, J.-J.: A dynamic programming approach to line segment matching in stereo vision. Pattern Recognition 27(8), 961–986 (1994)
18. Lee, J.J., Shim, J.C., Ha, U.H.: Stereo correspondence using the Hopfield neural network of a new energy function. Pattern Recognition 27(11), 1513–1522 (1994)
19. Mortara, M., Spagnuolo, M.: Similarity measures for bending polygonal shapes. Computer & Graphics 35, 13–27 (2001)
20. Moravec, H.: Robot Rover Visual Navigation. U.M.I. Research Press, Ann Arbor (1981)
21. Mousavi, M.S., Schalkoff, R.J.: Stereo vision: a neural network application to constraint satisfaction problem. In: SPIE 1382 Intelligent Robots and Computer Vision IX: Neural, Biological, and 3-D Methods, pp. 228–239 (1990)
22. Mousavi, M.S., Schalkoff, R.J.: A parallel distributed algorithm for feature extraction and disparity analysis of computer Images. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 428–435 (1990)
23. Nasrabadi, N.M., Choo, C.Y.: Hopfield network for stereo vision correspondence. IEEE Transactions on Neural Networks 3(1), 5–13 (1992)
24. Nasrabadi, N.M., Li, W.: Object recognition by a Hopfield neural network. IEEE Transactions on Systems, Man, and Cybernetics 21(6), 1523–1535 (1991)
25. Ohta, Y., Kanade, T.: Stereo by intra- and-inter-scanline search using dynamic programming. IEEE Transactions on Pattern Analysis and Machine Intelligence 7(2), 139–154 (1985)
26. Pajares, G., Cruz, J.M., Aranda, J.: Relaxation by Hopfield network in stereo image matching. Pattern Recognition 31(5), 561–574 (1997)

27. Ruicheck, Y., Postaire, J.-G.: A neural implementation for high speed processing in linear stereo vision. In: 1995 IEEE International Conference on Systems, Man and Cybernetics, vol. 5, pp. 3902–3907 (1995)
28. Parvin, B., Medioni, G.: A layered network for the correspondence of 3D objects. In: Proceedings of the 1991 IEEE International Conference on Robotics and Automation, Sacramento, California, pp. 1808–1813 (1991)
29. Shirai, Yoshiaki: Three-Dimensional Computer Vision. Springer, Heidelberg (1987)
30. Tank, D., Hopfield, J.: Simple 'neural' optimization networks: an A/D converter, signal decision circuit, and a linear programming circuit. IEEE Transactions on Circuits and Systems CAS 33(5) (1986)
31. Wolfe, W.J.: A parallel approach to simultaneously solving the correspondence problem and the pose estimation problem. In: Proceeding of the SPIE Conference on Mobile Robots (VI), vol. 1613, pp. 120–126 (1991)
32. Wolfe, W.J., Magee, M.: Fusion of multiple views of multiple reference points using a parallel distributed processing approach. In: Proceeding of the SPIE Conference on Sensor Fusion (III), vol. 1383. SPIE (1990)
33. Zurada, J.M.: Introduction to artificial neural systems. West Publishing Company, New York (1992)

Monotonic Monitoring of Discrete-Event Systems with Uncertain Temporal Observations

Gianfranco Lamperti and Marina Zanella

Dipartimento di Elettronica per l'Automazione
Via Branze 38, 25123 Brescia, Italy
{lamperti,zanella}@ing.unibs.it

Abstract. In discrete-event system monitoring, the observation is fragmented over time and a set of candidate diagnoses is output at the reception of each fragment (so as to allow for possible control and recovery actions). When the observation is uncertain (typically, a DAG with partial temporal ordering) a problem arises about the significance of the monitoring output: two sets of diagnoses, relevant to two consecutive observation fragments, may be unrelated to one another, and, even worse, they may be unrelated to the actual diagnosis. To cope with this problem, the notion of monotonic monitoring is introduced, which is supported by specific constraints on the fragmentation of the uncertain temporal observation, leading to the notion of stratification. The paper shows that only under stratified observations can significant monitoring results be guaranteed.

Keywords: Monitoring, Diagnosis, Discrete-event system, Uncertain temporal observation.

1 Introduction

Model-based diagnosis of discrete-event systems (DESs) has arisen a great interest in the last decade [3,9,2,10]. A DES consists of several interconnected components, where the favorite formalism to represent the behavior of each component is an automaton. Interconnections between components can either be modeled through explicit *ad hoc* primitives, called *links* [1], and/or implicitly, that is, a communication buffer may be indistinguishable from a component [10]. Distinct approaches to diagnosis of DESs in the literature can either assume that several state changes of distinct components can occur simultaneously [13,14], or not [1,8]. A diagnosis task requires an observation as input. Therefore, observation features and models have been investigated [5,4]. An observation is *temporally uncertain* if the generation order of observed events is not precisely known, what is known is instead a partial order that conforms to the actual generation order. In other words, an event can be observed before another that was generated by the DES before it, and, given the reception order of events, it is impossible to devise the relative emission order of all the pairs of events belonging to the observation. Therefore, several sequences of observable events comply with a temporally uncertain observation. Two diagnostic tasks inherent to DESs can be singled out: a-posteriori diagnosis [8], and monitoring-based diagnosis [12,11,6]. The former finds out the faults affecting a DES in an off-line way with respect to the system, by typically processing

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 348–362, 2009.
© Springer-Verlag Berlin Heidelberg 2009

the whole observation relevant to a complete evolution of the system. The latter, instead, tries and follow the evolution of a DES while it is occurring. The claim of this paper is that the criterion of soundness and completeness of results does not suffice for monitoring-based diagnosis in case the considered observation is affected by temporal uncertainty since it does not guarantee an important property that we call *monotonicity*. Such a property consists in producing as output at each monitoring step a set of diagnoses that includes the actual diagnosis. This paper shows how monotonicity depends not only on the abilities of the problem-solving method but also on the characteristics of the considered observation, and discusses the granularity with which a temporally uncertain observation has to be processed by a sound and complete problem-solving method so that diagnostic results are monotonic, whichever the DES.

2 Discrete-Event Systems

All the possible evolutions over time of a DES (cumulatively called the *global system behavior*) can be thought of as the paths of a directed graph, where each node is a *system state* (this being the composition of the states of all components and explicit links, if any) and each arc is a system state change, called a *transition*. This is a conceptual standpoint, not an operational one, in that most current approaches in the literature need not generate any global system behavior. In other words, approaches to diagnosis of DESs can reason about all the feasible evolutions of a DES without generating them, by just exploiting the individual behaviors of components and connections, the assumptions about the (either simultaneous or not) state change triggering, and (possibly) also domain dependent constraints. The reason for we take the global system behavior into consideration is that it establishes a common ground for defining the outputs of any diagnosis task inherent to DESs, independently of any specific (modeling and processing) approach. Formally, given a system Σ and an initial state Σ_0, each evolution of Σ is confined within the *behavior space*, $Bhv(\Sigma, \Sigma_0)$. The latter is a directed graph rooted in Σ_0, where each node is a state of Σ and each arc is a transition. Each path within the behavior space is a *history* of Σ. As such, a history is a (possibly empty) sequence of transitions rooted in Σ_0.

Example 1. Fig. 1(a) draws the behavior space of a DES called $\bar{\Sigma}$. Nodes represent system states, where 0 is the initial state. Each arc represents a state change and is marked by a transition identifier. A (possibly cyclic) path rooted in 0 is a history of $\bar{\Sigma}$, for instance, $\bar{h} = \langle X_1, X_2, Y_2, Z_4, Z_3, Y_4, W_4, Z_2, X_1 \rangle$.

Since the adopted conceptual representation of the global system behavior is quite general and common to distinct approaches to model-based diagnosis of DESs (and, therefore, distinct classes of DESs) in the literature, it is pointless to distinguish whether each transition is an individual component state change or not. For instance, X_i, Y_i, W_i, and Z_i, $i \in [1\,..\,4]$, might be the completely asynchronous transitions of a distributed DES featuring four components X, Y, W, and Z. Or X_i, Y_i, and W_i might be the asynchronous transitions of three components while each transition Z_i might be triggered simultaneously in the three components altogether. In either case what is dealt with in the next sections is the same.

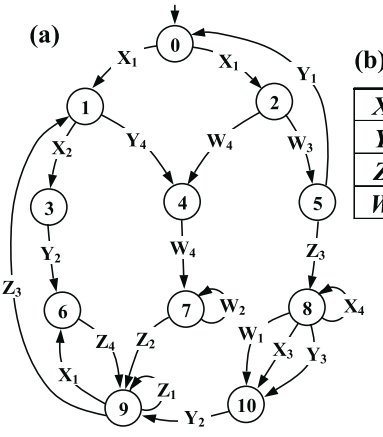


Fig. 1. Behavior space $Bhv(\bar{\Sigma}, \bar{\Sigma}_0)$ (a), and viewer and ruler matrix (b) for $\bar{\mathcal{V}}$ and $\bar{\mathcal{R}}$

3 Diagnosis

A-posteriori diagnosis finds out the faults affecting a DES, given a relevant observation and the initial state of the system. The system, starting from its initial state, is assumed to have undergone a sequence of state transitions, some of which are associated with an observable event and/or a fault. We call *actual history* the sequence of transitions actually followed by the system, and *actual diagnosis* the set of faults entailed by the actual history. The output of the a-posteriori diagnosis task is a set of sound candidate diagnoses, with each diagnosis being a set of faults. A candidate is *sound* if it is entailed by a history that generates a sequence of observable events that comply with the given observation. The set of histories inherent to an observation is complete if it encompasses all and only the paths, included in the global system behavior, that start from the given initial state and produce a sequence of observable events that comply with the observation. The complete set of histories entails the *complete* set of candidate diagnoses. Formally, the observability and abnormality properties of a DES can be represented as follows. Let $\mathbf{T}$ be the domain of transitions in Σ and $\mathbf{L}_o$ a domain of *observable labels*. A *viewer* $\mathcal{V}$ of Σ is a function from $\mathbf{T}$ to $(\mathbf{L}_o \cup \{\epsilon\})$, where ϵ is the *null* label. If $(T, \epsilon) \in \mathcal{V}$ then T is *silent* else T is *visible*. $\mathcal{V}$ is said to cause *source uncertainty* when it includes two pairs (T_1, ℓ) and (T_2, ℓ) where $T_1 \neq T_2$. Let h be a history of Σ, the *trace* $h \boxtimes \mathcal{V}$ is the sequence of observable labels relevant to h, $h \boxtimes \mathcal{V} = \langle \ell \mid T \in h, (T, \ell) \in \mathcal{V}, \ell \neq \epsilon \rangle$.

Example 2. Assuming $\mathbf{L}_o = \{a, b, c, d\}$, a viewer $\bar{\mathcal{V}}$ for $\bar{\Sigma}$ is defined by the following associations, displayed in the white cells of the matrix of Fig. 1(b), with the other transitions being silent: (X_1, a), (Y_2, b), (Z_3, c), (W_4, d). Based on $\bar{\mathcal{V}}$, the trace of history $\bar{h}$ defined in Example 1 is $\bar{h} \boxtimes \bar{\mathcal{V}} = \langle a, b, c, d, a \rangle$.

Ideally, the trace $h \boxtimes \mathcal{V}$ should represent how h is observed outside Σ. However, what is actually perceived is the *observation* $\mathcal{O}$, $\mathcal{O} = \mathcal{U}(h \boxtimes \mathcal{V})$. $\mathcal{U}$ is a (nondeterministic unknown) *uncertainty function* that generates a directed acyclic graph (DAG),

$\mathcal{O} = (\mathcal{N}, \mathcal{A})$, where $\mathcal{N}$ is the set of nodes and $\mathcal{A}$ the set of arcs, with the following *uncertainty properties*:

- (*Logical uncertainty*) Each label ℓ in the trace corresponds to a node in $\mathcal{O}$; such a label is perceived as a subset of $(\mathbf{L}_o \cup \{\epsilon\})$ of *candidate labels*, necessarily including ℓ;
- (*Node uncertainty*) Additional (spurious) nodes are possibly inserted into $\mathcal{O}$, each of which is associated with a subset of candidate labels necessarily including ϵ;
- (*Temporal uncertainty*) Absolute temporal ordering of the trace is relaxed to partial ordering (with the latter being consistent with the former).

The uncertainty function of a system is not given once and for all; instead it depends on the specific setting of the system at hand (sensors, channels conveying sensor values to the observer, etc.) Since such a function is nondeterministic, distinct instances of the same system run may be perceived by the same observer as different observations, represented by distinct DAGs, possibly including a different number of nodes and arcs. The *extension* of a node N in $\mathcal{N}$, written $\|N\|$, is the set of labels embodied in N. A *candidate trace* of $\mathcal{O}$ is a sequence of labels obtained by first picking up a label from each $\|N\|$, $N \in \mathcal{N}$, without violating the ordering imposed by $\mathcal{A}$, and then removing the ϵ labels from the sequence. The *extension* of $\mathcal{O}$, $\|\mathcal{O}\|$, is the whole set of candidate traces of $\mathcal{O}$. If $\mathcal{O}$ is the observation relevant to a history h, then the trace relevant to h is among the candidate traces of $\mathcal{O}$, as claimed by Proposition 1.

Proposition 1. *If* $\mathcal{O} = \mathcal{U}(h \boxtimes \mathcal{V})$ *then* $(h \boxtimes \mathcal{V}) \in \|\mathcal{O}\|$.

Proof. Let $\mathcal{O} = (\mathcal{N}, \mathcal{A})$. Let $h_v = \langle T_1, T_2, .., T_n \rangle$ be the sequence of visible transitions in h, that is, $h_v = \langle T_i \in h \mid (T_i, \ell_i) \in \mathcal{V}, \ell_i \neq \epsilon \rangle$. Therefore, based on the notion of logical uncertainty, $\mathcal{N}$ consists of n nodes N_i, $i \in [1 \,..\, n]$, each corresponding to a label in the trace $h \boxtimes \mathcal{V}$, and, based on the notion of node uncertainty, $\mathcal{N}$ possibly includes a number of additional spurious nodes, each containing ϵ. Let $p = \langle N_1, N_2, .., N_n \rangle$ be the (only) permutation of the above n nodes N_i in $\mathcal{N}$ that has the absolute temporal ordering of the trace of h. Hence, there exists a bijective mapping between h_v and p, according to which T_i corresponds to N_i and vice versa. According to the notion of temporal uncertainty, the ordering of p complies with $\mathcal{A}$. Now we generate a candidate trace s of $\mathcal{O}$ by picking up from each $N_i \in \mathcal{N}$, without violating the ordering imposed by p, the label ℓ_i generated by T_i, and from each spurious node in $\mathcal{N}$ the ϵ label. Note that, according to the definition of logical uncertainty, $\ell_i \in \|N_i\|$. Hence s equals the trace of h, $s = h \boxtimes \mathcal{V}$, and, at the same time, being a candidate trace, it belongs to the extension of $\mathcal{O}$, that is, $s \in \|\mathcal{O}\|$. □

Example 3. Based on $\bar{h}$ and $\bar{\mathcal{V}}$, depicted in Fig. 2 is an observation $\bar{\mathcal{O}} = \mathcal{U}(\bar{h} \boxtimes \bar{\mathcal{V}})$ of $\bar{\Sigma}$. Note how logical uncertainty holds in node N_4, where $\|N_4\| = \{d, \epsilon\}$, and temporal uncertainty involves nodes N_2 and N_3, whose reciprocal emission order is unknown. $\bar{\mathcal{O}}$ is not affected by node uncertainty, as the number of nodes equals the length of the trace of $\bar{h}$. The extension of $\bar{\mathcal{O}}$ is[1] $\|\bar{\mathcal{O}}\| = \{abca, acba, abcda, acbda\}$ which, in accordance with Proposition 1, includes $\bar{h} \boxtimes \bar{\mathcal{V}} = abcda$.

[1] For convenience, candidate traces are written as strings of labels.

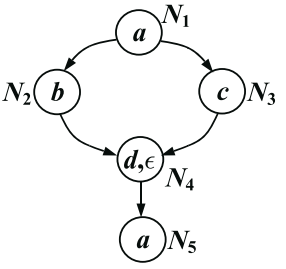


Fig. 2. Observation $\bar{\mathcal{O}}$ for $\bar{\Sigma}$

Like a viewer, we can define a mapping $\mathcal{R}$, the *ruler* of Σ, from $\mathbf{T}$ to $(\mathbf{L}_\mathrm{f} \cup \{\epsilon\})$, where $\mathbf{L}_\mathrm{f}$ is a set of *fault labels*. If $(T, \epsilon) \in \mathcal{R}$ then T is *normal* else T is *faulty*. Given $\mathcal{R}$, the *diagnosis* $h \otimes \mathcal{R}$ is the set of fault labels

$$h \otimes \mathcal{R} = \{\ell \mid T \in h, (T, \ell) \in \mathcal{R}, \ell \neq \epsilon\}. \tag{1}$$

A diagnosis is empty when all transitions in h are normal.

Example 4. Assuming $\mathbf{L}_\mathrm{f} = \{x, y, z, w\}$, a ruler $\bar{\mathcal{R}}$ for $\bar{\Sigma}$ is defined by the following associations displayed in the gray cells of the matrix of Fig. 1(b), with the other transitions being normal: (X_2, x), (Y_4, y), (Z_2, z), (W_3, w). Based on $\bar{\mathcal{R}}$, the diagnosis of history $\bar{h}$ defined in Example 1 is $\bar{\delta} = \bar{h} \otimes \bar{\mathcal{R}} = \{x, y, z\}$.

An *(a-posteriori) diagnosis problem* relevant to a system Σ is a 4-tuple involving an initial state, a viewer, an observation, and a ruler,

$$\wp(\Sigma) = (\Sigma_0, \mathcal{V}, \mathcal{O}, \mathcal{R}). \tag{2}$$

The *solution* of $\wp(\Sigma)$, written $\Delta(\wp(\Sigma))$, consists of a set of *candidate diagnoses*, with each diagnosis being entailed by a history h in $Bhv(\Sigma, \Sigma_0)$ whose trace conforms to $\mathcal{O}$. As such, $\Delta(\wp(\Sigma))$ is

$$\{\delta \mid \delta = h \otimes \mathcal{R}, h \in Bhv(\Sigma, \Sigma_0), h \boxtimes \mathcal{V} \in \|\mathcal{O}\|\}. \tag{3}$$

The set of candidate diagnoses defined in (3) is sound and complete; in addition, it includes the diagnosis relevant to the actual history of Σ, as claimed by Proposition 2.

Proposition 2. *Let $\wp(\Sigma) = (\Sigma_0, \mathcal{V}, \mathcal{O}, \mathcal{R})$ be a diagnosis problem, where $\mathcal{O} = \mathcal{U}(h \boxtimes \mathcal{V})$. Then, $\delta \in \Delta(\wp(\Sigma))$, where $\delta = h \otimes \mathcal{R}$.*

Proof. By the definition of behavior space, the actual history h of the system that caused the diagnostic problem $\wp(\Sigma)$ belongs to $Bhv(\Sigma, \Sigma_0)$. Owing to Proposition 1, its trace belongs to the extension of $\mathcal{O}$, that is, $h \boxtimes \mathcal{V} \in \|\mathcal{O}\|$. Therefore, owing to the definition of the set of candidate diagnoses provided by Eq. (3), $\Delta(\wp(\Sigma))$ includes diagnosis δ, where, as assumed $\delta = h \otimes \mathcal{R}$. □

If h is the (unknown) actual history of Σ, then $\Delta(\wp(\Sigma))$ includes (besides the actual diagnosis δ) a (possibly empty) set of *spurious diagnoses*, each of which is entailed by (at least) one history $h' \neq h$ such that $(h' \boxtimes \mathcal{V}) \in \|\mathcal{O}\|$.

Example 5. Let $\wp(\bar{\Sigma}) = (\Sigma_0, \bar{\mathcal{V}}, \bar{\mathcal{O}}, \bar{\mathcal{R}})$. As already remarked, a matrix representing both viewer $\bar{\mathcal{V}}$ and ruler $\bar{\mathcal{R}}$ is drawn in Fig. 1(b). Since, in this example, faulty transitions are silent, each matrix element contains one label at most, either observable or faulty, where faulty labels are shaded. The solution of $\wp(\bar{\Sigma})$ includes the diagnosis relevant to history $\bar{h}$, namely $\bar{\delta} = \{x, y, z\}$, and the spurious diagnosis $\bar{\delta}' = \{w\}$. In fact, there exist histories other than $\bar{h}$, for instance, $\bar{h}' = \langle X_1, W_3, Z_3, X_3, Y_2, X_1\rangle$, entailing $\bar{\delta}'$, whose trace $acba$ belongs to $\|\bar{\mathcal{O}}\|$.

4 Monitoring

Monitoring-based diagnosis finds out the faults affecting a DES iteratively, once for each considered chunk of an observation, as soon as such a chunk has been received. Thus, solving a monitoring-based diagnosis problem can be seen as repeatedly solving a-posteriori diagnosis problems, where the a-posteriori diagnosis problem solved at time $t+1$ is inherent to the observation considered at time t plus the observation chunk received in the meantime from t to $t+1$. However, from the operational point of view, the methods for facing the two classes of problems are different since in monitoring-based diagnosis the problem at time $t+1$ is solved by exploiting the solution of the problem at time t.

The quality of the results produced by an a-posteriori diagnosis method is usually assessed based on their soundness and completeness. In fact, if all the outputs are actually candidates and no candidate is missing, this criterion guarantees that the (only) actual diagnosis is included in the set of candidates provided as output. This conclusion, however, holds only under the condition that the observation is complete, that is, all the observable events inherent to the considered time interval have already been received. Such a condition, in general, is not fulfilled during monitoring, wherein some observable events included in the observation chunk received by time $t+1$ may have been emitted before some events included in the previously received chunks.

The uncertain observation taken as input by a monitoring task can still be represented as a DAG. However, this DAG is not given as a whole but, rather, as a list of several *fragments*, where each fragment is composed of one or several nodes along with the relevant temporal constraints (arcs).

Formally, let $\mathcal{O} = (\mathcal{N}, \mathcal{A})$. A *fragmentation* of $\mathcal{O}$ is a sequence $\mathcal{O}^* = \langle F_1, \ldots, F_n\rangle$, where each fragment $F_i = (\mathcal{N}_i, \mathcal{A}_i)$, $i \in [1\,..\,n]$, is such that $\{\mathcal{N}_1, \ldots, \mathcal{N}_n\}$ and $\{\mathcal{A}_1, \ldots, \mathcal{A}_n\}$ are partitions of $\mathcal{N}$ and $\mathcal{A}$, respectively. Each fragment F_i represents a set of observable events received in the current time interval. Each node in $\mathcal{N}_i$ is an event and each arc in $\mathcal{A}_i$ is a temporal precedence relationship (according to the emission order). In particular, $\mathcal{A}_i$ is the minimal set of precedence relationships linking the observable events received in the current interval with each other and with the other events belonging to the observation. In other words, $\mathcal{A}_i$ includes all and only the relationships linking the nodes in $\mathcal{N}_i$ with their parent nodes.

That given above is the most general definition of a fragmented observation, since it does not constrain the set of parent nodes of a given node in the observation DAG. This means that an event received in the current interval may have one or more parent events (i.e. events that precede it in the emission order) that have not been received yet. Such a

freedom degree can be suppressed by imposing the following condition, which requires the parents of nodes in the new fragment to be in the fragments received up to now:

$$\forall N \mapsto N' \in \mathcal{A}_i \left(N' \in \mathcal{N}_i, N \in \bigcup_{j=1}^{i} \mathcal{N}_j \right). \tag{4}$$

Each nonempty prefix $\langle F_1, \ldots, F_i \rangle$ of $\mathcal{O}^*$ corresponds to a *sub-observation* $\mathcal{O}_{[i]} = (\mathcal{N}_{[i]}, \mathcal{A}_{[i]})$, where

$$\mathcal{N}_{[i]} = \bigcup_{j=1}^{i} \mathcal{N}_j \qquad \mathcal{A}_{[i]} = \bigcup_{j=1}^{i} \mathcal{A}_j \tag{5}$$

The *empty* sub-observation is $\mathcal{O}_{[0]} = (\emptyset, \emptyset)$. If $\mathcal{O}$ is known, $\mathcal{O}^*$ is univocally defined by the sequence of $\mathcal{N}_i$, as (4) entails that $\mathcal{A}_i$ necessarily includes all (and only) the arcs entering nodes in $\mathcal{N}_i$. For each $i \in [0\,..\,n]$, we can define a *sub-problem* $\wp_{[i]}(\Sigma) = (\Sigma_0, \mathcal{V}, \mathcal{O}_{[i]}, \mathcal{R})$.

Example 6. A fragmentation $\bar{\mathcal{O}}^*$, where $\bar{\mathcal{O}}$ is displayed in Fig. 2, is defined by the following sequence of sets of nodes: $\langle \{N_1, N_3\}, \{N_2\}, \{N_4\}, \{N_5\} \rangle$.

A *monitoring problem* is a 4-tuple involving an initial state, a viewer, a fragmented observation, and a ruler,

$$\mu(\Sigma) = (\Sigma_0, \mathcal{V}, \mathcal{O}^*, \mathcal{R}). \tag{6}$$

The *solution* of $\mu(\Sigma)$ is the sequence of the solutions of the diagnosis sub-problems $\wp_{[i]}(\Sigma)$, $i \in [0\,..\,n]$,

$$\Delta(\mu(\Sigma)) = \left\langle \Delta(\wp_{[0]}(\Sigma)), \ldots, \Delta(\wp_{[n]}(\Sigma)) \right\rangle. \tag{7}$$

Example 7. Let $\mu(\bar{\Sigma}) = (\Sigma_0, \bar{\mathcal{V}}, \bar{\mathcal{O}}^*, \bar{\mathcal{R}})$ be a monitoring problem inherent to the evolution described in Example 1. Then, $\Delta(\mu(\bar{\Sigma})) = \langle \Delta_0, \Delta_1, \Delta_2, \Delta_3, \Delta_4 \rangle$, where $\Delta_0 = \{\emptyset\}$, $\Delta_1 = \{\{w\}\}$, $\Delta_2 = \{\{w\}, \{x\}, \{x, y\}\}$, $\Delta_3 = \{\{w\}, \{x\}, \{x, y\}, \{x, y, z\}\}$, and $\Delta_4 = \{\{w\}, \{x, y, z\}\}$.

Example 7 shows that the solution of a monitoring problem, although consisting in a sound and complete set of diagnoses, is disappointing. In fact, at monitoring step 1, one is induced to believe that w is a quite certain fault but, from iteration 2 on, fault w is not certain any more. The rationale behind this deceitful behavior is that any sound and complete set of outputs complies with the whole observation received so far as it were a complete observation, while it is not. Therefore, the extension of the observation may change non-monotonically from one step to another, thus producing the highlighted negative effect.

Example 8. In Example 7 the histories consistent with the (only) candidate trace ac of $\mathcal{O}_{[1]}$ are those belonging to the graph displayed in Fig. 3(a), where the final states are double circled and the relevant label (shaded if faulty) is associated with each arc. The histories consistent with $\mathcal{O}_{[2]}$, displayed in Fig. 3(b), are those producing either

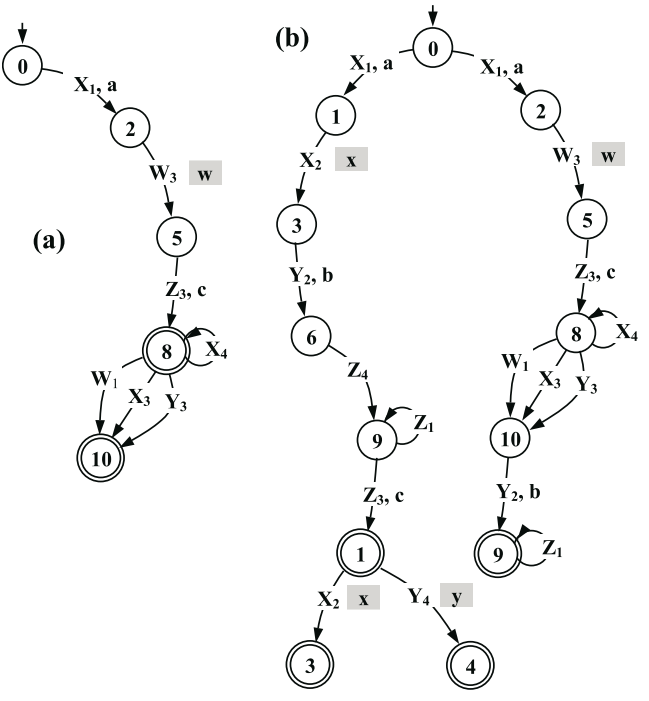


Fig. 3. Nonmonotonic update of histories

the candidate trace acb or abc. Fig. 3(b) includes a left subgraph that causes two new diagnoses to be added in Δ_2 with respect to Δ_1. This subgraph has been completely added since no observation prefix of abc had been considered in the previous step. This is a consequence of the fact that nodes N_2 and N_3, whose reciprocal temporal order is unknown, have been considered in two different steps since they belong to two distinct fragments. Note that a candidate diagnosis output at a certain time step, even if the only one, may be completely refuted in subsequent steps. This would have occurred, for instance, to candidate diagnosis $\{w\}$ at the second step of our example if there were no histories in the global behavior space producing the sequence of observable events acb.

The negative effect we have highlighted is not a consequence of the restriction imposed on the definition of a fragmented observation by Condition (4). In fact, if such a condition does not hold, the effect is amplified. Therefore, we keep on considering the notion of a fragmented observation that fulfills (4).

4.1 Stratification

Given a monitoring context, we call *monotonicity* the property of producing as output at each iteration a set of candidates that includes the actual candidate. Monotonicity is an essential property in monitoring: results that are not monotonic are hardly

useful and dependable. Let $\mu(\Sigma) = (\Sigma_0, \mathcal{V}, \mathcal{O}^*, \mathcal{R})$, where $\mathcal{O}^* = \langle F_1, \ldots, F_n \rangle$. Let $\langle \Delta_0, \Delta_1, \ldots, \Delta_n \rangle$ be the solution of $\mu(\Sigma)$ and δ the actual (unknown) diagnosis of the actual (unknown) history of Σ. Since $\Delta_n = \Delta(\wp_{[n]}(\Sigma)) = \mathrm{ThestratificationinExample}\ \Delta(\wp(\Sigma))$, based on Proposition 2, we have $\delta \in \Delta_n$. We say that $\mu(\Sigma)$ is *monotonic* iff $\forall i \in [0\,..\,(n-1)]$ there exists $\delta_i \in \Delta_i$ such that

$$\delta_0 \subseteq \delta_1 \subseteq \ldots \subseteq \delta_{n-1} \subseteq \delta. \tag{8}$$

Example 9. The monitoring problem $\mu(\bar{\Sigma})$ in Example 7 is *not* monotonic: the actual diagnosis is $\bar{\delta} = \{x, y, z\}$, for which Condition (8) does not hold as $\Delta_1 = \{\{w\}\}$ includes no diagnosis that is a subset of $\bar{\delta}$.

Interestingly, the monotonicity of a monitoring problem $\mu(\Sigma)$ depends on the nature of the fragmentation of $\mathcal{O}$. On the one hand, not all fragmentations $\mathcal{O}^*$ make $\mu(\Sigma)$ monotonic. On the other, the *trivial fragmentation* involving the whole observation $\mathcal{O}$ as the unique fragment supports monotonicity, but this is in fact a-posteriori diagnosis, not monitoring. Thus, we are interested in the nature of nontrivial fragmentations that guarantee monotonicity, independently of the specific system (and behavior space) at hand, namely nontrivial *stratified observations*.

A fragmentation $\mathcal{O}^* = \langle F_1, \ldots, F_n \rangle$ is *stratified* iff for each fragment $F_i = (\mathcal{N}_i, \mathcal{A}_i)$, $i \in [1\,..\,n]$, we have

$$\forall N \in \mathcal{N}_i \ \ (Unrl(N) \subseteq \ \mathcal{N}_i) \tag{9}$$

where $Unrl(N)$ is the set of all the nodes in $\mathcal{N}$ whose reciprocal emission order with respect to N is unknown. A stratified fragmentation is called a *stratification* and each fragment a *stratum*. Condition (9) requires that all nodes that are neither ancestors nor descendants (namely, unrelated) of nodes in the i-th stratum, be in the i-th stratum themselves. This allows candidate traces of sub-observations to grow incrementally (monotonically), as claimed by Proposition 3.

Proposition 3. *Let $\mathcal{O}^* = \langle F_1, \ldots, F_n \rangle$ be a stratified observation. Then, for each $i \in [1\,..\,n]$, $\|\mathcal{O}_{[i]}\|$ is composed of all the traces in $\|\mathcal{O}_{[i-1]}\|$ (possibly) extended with further observable labels.*

Proof. If the fragmentation is trivial, $\mathcal{O}^* = \mathcal{O}_{[1]} = \mathcal{O} = \langle F_1 \rangle$, then $\|\mathcal{O}_{[1]}\|$ includes all the candidate traces of $\mathcal{O}$. Since, for every fragmented observation $\|\mathcal{O}_{[0]}\| = \{\epsilon\}$, and each trace in $\|\mathcal{O}_{[1]}\|$ either equals or extends the null trace, the proposition is proved for $n = 1$ (and $i = 1$). If $n > 1$, let $F_i = (\mathcal{N}_i, \mathcal{A}_i)$, $i \in [1\,..\,n]$. Let $\|F_i\|$ be the *stratum extension* of F_i, which is the set of all the *trace postfixes* of F_i. A trace postfix of a stratum F_i is a sequence of labels in $\mathbf{L}_o$ obtained by picking up a label from each $\|N\|$, $N \in \mathcal{N}_i$, without violating the ordering imposed by $\mathcal{A}_i$. Every node $N \in \mathcal{N}_i$ cannot be completely disconnected from nodes in the higher strata (that is, nodes belonging to $\bigcup_{j=1}^{i-1} \mathcal{N}_j$) since N would be unrelated with respect to the nodes in the higher strata, which contradicts the hypothesis, according to which F_i is a stratum and, as such, all the unrelated nodes of the nodes in the i-th stratum are in the i-th stratum themselves. In order not to be unrelated with the nodes in the higher strata, it is necessary that $\forall N \in \mathcal{N}_i, \exists N' \mapsto N \in \mathcal{A}_i$, $N' \notin \mathcal{N}_i$. Eq. (4) prevents the existence of

any $N' \mapsto N \in \mathcal{A}_i$ such that N' belongs to lower strata than F_i. This implies that, $\forall N \in \mathcal{N}_i, \exists N' \mapsto N \in \mathcal{A}_i$, $N' \in \bigcup_{j=1}^{i-1}\mathcal{N}_j$. Moreover, each such N' cannot be the parent node of any other node N'', $N'' \in \bigcup_{j=1}^{i-1}\mathcal{N}_j$, since this would imply that N is unrelated with respect to N'', which contradicts the hypothesis, according to which F_i is a stratum and, as such, all the unrelated nodes of N belongs to $\mathcal{N}_i$. Therefore, any N' is bound to be a *leaf* node of a higher stratum, this being a node which is not the source of any arc in $\bigcup_{j=1}^{i-1}\mathcal{A}_j$. However, there cannot be any leaf node in all the strata F_j, $j \in [1\,..\,i-2]$, since such a leaf node would be unrelated with the nodes in strata F_s, $s \in [j+1\,..\,i-1]$, which contradicts the hypothesis. Therefore, any N' is bound to be a *leaf* node of F_{i-1}. On the other hand, any leaf node of F_{i-1} is bound to be the parent node of a node in F_i. In fact, a leaf node of F_{i-1} cannot be a final node of the whole observation, since final nodes are bound to belong to the last stratum (if they belonged to another stratum, they would be unrelated with the nodes in the lower strata, which contradicts the stratification hypothesis). Analogously, a leaf node of F_{i-1} cannot be the parent node of any node belonging to strata F_k, $k > i$, since it would be unrelated with respect to all the nodes belonging to the strata F_r, $r \in [i\,..\,k-1]$, which contradicts the stratification hypothesis. Then, all the leaf nodes of F_{i-1} have a child node at least in F_i, and all the nodes of F_i have a parent node at least in F_{i-1}. Therefore, each trace in $\|\mathcal{O}_{[i]}\|$, is obtained by extending a trace in $\|\mathcal{O}_{[i-1]}\|$, relevant to a path ending in leaf node N' of F_{i-1}, with a trace postfix of F_i, relevant to a path beginning in a node N of F_i that is a child node of N'. This way, all the trace postfixes of F_i are exploited. Since there may be trace postfixes that equal the null sequence, each trace in $\|\mathcal{O}_{[i]}\|$ either is an extension of a trace in $\|\mathcal{O}_{[i-1]}\|$ or equals a trace in $\|\mathcal{O}_{[i-1]}\|$, which proves the thesis. □

Example 10. Fragmentation $\bar{\mathcal{O}}^*$ defined in Example 6 is *not* a stratification as the first stratum $\{N_1, N_3\}$ does not include N_2, the unrelated node of N_3. A possible stratification of $\bar{\mathcal{O}}$ is: $\langle\{N_1\}, \{N_2, N_3\}, \{N_4\}, \{N_5\}\rangle$. As expected by Proposition 3, the traces of the relevant sub-observations grow monotonically, in fact $\|\bar{\mathcal{O}}_{[0]}\| = \{\epsilon\}$, $\|\bar{\mathcal{O}}_{[1]}\| = \{a\}$, $\|\bar{\mathcal{O}}_{[2]}\| = \{abc, acb\}$, $\|\bar{\mathcal{O}}_{[3]}\| = \{abc, acb, abcd, acbd\}$, and $\|\bar{\mathcal{O}}_{[4]}\| = \{abca, acba, abcda, acbda\}$.

Proposition 3 states a conceptual property, while it does not imply in any way that the extensions of the sub-observations of a stratified observation be computed in order to solve a monitoring problem. The incremental growing of candidate traces entails monotonic monitoring, as stated by Proposition 4.

Proposition 4. *A monitoring problem with stratified observation is monotonic.*

Proof. Let $\mu(\Sigma) = (\Sigma_0, \mathcal{V}, \mathcal{O}^*, \mathcal{R})$ be a monitoring problem involving the stratified observation $\mathcal{O}^* = \langle F_1, \ldots, F_n\rangle$. According to Eq. (7), the solution is $\Delta(\mu(\Sigma)) = \big\langle \Delta(\wp_{[0]}(\Sigma)), \ldots, \Delta(\wp_{[n]}(\Sigma))\big\rangle$, where $\wp_{[i]}(\Sigma) = (\Sigma_0, \mathcal{V}, \mathcal{O}_{[i]}, \mathcal{R})$, $i \in [0\,..\,n]$. Based on Eq. (3), $\Delta(\wp_{[i]}) = \{\delta_i \mid \delta_i = h_i \otimes \mathcal{R}, h_i \in Bhv(\Sigma, \Sigma_0), h_i \boxtimes \mathcal{V} \in \|\mathcal{O}_{[i]}\|\}$. According to Proposition 3, $\forall i \in [1\,..\,n]$, $\|\mathcal{O}_{[i]}\|$ is composed of all the traces in $\|\mathcal{O}_{[i-1]}\|$ (possibly) extended with further observable labels. Therefore, each $h_i \in Bhv(\Sigma, \Sigma_0)$, $h_i \boxtimes \mathcal{V} \in \|\mathcal{O}_{[i]}\|$ is a (possible) extension of a history $h_{i-1} \in Bhv(\Sigma, \Sigma_0)$, $h_{i-1} \boxtimes \mathcal{V} \in \|\mathcal{O}_{[i-1]}\|$. Consequently, each diagnosis $\delta_i = h_i \otimes \mathcal{R}$ is a superset of a diagnosis

$\delta_{i-1} = h_{i-1} \otimes \mathcal{R}$, where h_i is an extension of h_{i-1}. Since, according to Eq. (3), $\delta_{i-1} \in \Delta(\wp_{[i-1]}(\Sigma))$, this proves that $\forall \delta_i \in \Delta_i$, $\exists \delta_{i-1} \in \Delta_{i-1}$ $(\delta_{i-1} \subseteq \delta_i)$. As already remarked, since $\Delta_n = \Delta(\wp_{[n]}(\Sigma)) = \Delta(\wp(\Sigma))$, based on Proposition 2, the actual diagnosis $\delta \in \Delta_n$. Therefore, $\exists \delta_{n-1} \in \Delta_{n-1}$ $(\delta_{n-1} \subseteq \delta)$. Analogously, $\exists \delta_{n-2} \in \Delta_{n-2}$ $(\delta_{n-2} \subseteq \delta_{n-1})$, and so on, that is, Condition (8) is fulfilled and, therefore, the proposition is proved. □

Example 11. Consider a variant $\mu'(\bar{\Sigma})$ of the monitoring problem defined in Example 7, where $\bar{\mathcal{O}}^*$ is replaced by the stratification introduced in Example 10. Then, $\Delta(\mu'(\bar{\Sigma})) = \langle \Delta'_0, \Delta'_1, \Delta'_2, \Delta'_3, \Delta'_4 \rangle$, where $\Delta'_0 = \{\emptyset\}$, $\Delta'_1 = \{\emptyset, \{w\}, \{x\}, \{y\}\}$, $\Delta'_2 = \{\ \{w\}, \{x\}, \{x, y\}\}$, $\Delta'_3 = \{\ \{w\}, \{x\}\ , \{x, y\}, \{x, y, z\}\}$, and $\Delta'_4 = \{\ \{w\}, \{x, y, z\}\}$. As expected by Proposition 4, $\mu'(\bar{\Sigma})$ is monotonic.

A nontrivial stratification $\mathcal{O}^*$ can always be transformed it into a different stratification by aggregating, for instance, contiguous strata in $\mathcal{O}^*$. In other words, property (9) is conserved when several contiguous strata are grouped together to form coarser-grained fragments. The contrary is not true: when two or more contiguous fragments are obtained by splitting a single stratum, stratification may be lost. We may be interested in considering the *finest stratification*, where strata cannot be further split without losing stratification. This allows the monitoring task to be as reactive as possible in generating candidate diagnoses.

Proposition 5. *The finest stratification is unique.*

Proof. Owing to Condition (4), a fragmentation is univocally identified by the sequence of $\mathcal{N}_i$ of its fragments. Therefore, a stratification, being a fragmentation whose fragments are called strata, is univocally identified by the sequence of $\mathcal{N}_i$ of its strata. The finest stratification $\mathcal{O}^*$ of a given observation $\mathcal{O} = (\mathcal{N}, \mathcal{A})$ is the one such that each stratum contains all and only the nodes that are bound to belong to the same stratum. We define the binary operator *same-stratum* $\diamondsuit$ between two nodes $N, N' \in \mathcal{N}$ this way: $N \diamondsuit N'$ iff it is necessary that N' belongs to the same stratum as N. The same-stratum relation is reflexive since, by definition, a fragmentation produces a partition of $\mathcal{N}$, where each fragment (a stratum in a stratification) is inherent to a part; therefore, each node $N \in \mathcal{N}$ necessarily belongs to just one stratum; in other words, N needs to belong to the same stratum as N itself. The same-stratum relation is symmetric. In fact, if $N' \in \mathit{Unrl}(N)$, then $N \diamondsuit N'$. $N' \in \mathit{Unrl}(N)$ means that $N' \notin \mathit{Anc}(N)$ and $N' \notin \mathit{Desc}(N)$, which imply $N \notin \mathit{Desc}(N')$ and $N \notin \mathit{Anc}(N')$, respectively. In other words, $N' \in \mathit{Unrl}(N)$ implies $N \in \mathit{Unrl}(N')$, where the latter entails $N' \diamondsuit N$, thus proving the symmetric property. The same-stratum relation is transitive. If $N' \in \mathit{Unrl}(N)$, then $N \diamondsuit N'$. Analogously, if $N'' \in \mathit{Unrl}(N')$, then $N' \diamondsuit N''$. If the two previous conditions hold, since Condition (9) has to be fulfilled for every node belonging to the same stratum, it follows that $N \diamondsuit N''$, which proves transitiveness. Hence, same-stratum partitions $\mathcal{N}$ into equivalence classes, each of which identifies a minimal stratum. □

Example 12. The stratification in Example 10 is in fact the finest one. By aggregating, for instance, the two intermediate strata, the new stratification $\langle \{N_1\}, \{N_2, N_3, N_4\}, \{N_5\} \rangle$ is obtained.

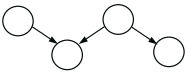


Fig. 4. Connected observation with trivial stratification

We may wonder whether all observations can be non-trivially stratified. The answer to this question comes from Proposition 6.

Proposition 6. *Given an observation $\mathcal{O}$ consisting of two nodes at least, a necessary (not sufficient) condition for a nontrivial stratification of $\mathcal{O}$ to exist is that $\mathcal{O}$ is connected.*

Proof. The proposition consists of two statements: (i) If $\mathcal{O}$ is connected, then a nontrivial stratification may exist or not; and (ii) If $\mathcal{O}$ is disconnected, its finest (and therefore only) stratification is the trivial stratification. Based on the proof of Proposition 5, the minimal strata of an observation $\mathcal{O} = (\mathcal{N}, \mathcal{A})$ are defined by the equivalence classes of nodes identified by the *same-stratum* relation, which is independent of the content of nodes. In case (i), assume that the observation consists just of a pair of nodes such as one is the parent of the other. These nodes belong to two distinct equivalence classes identified by same-stratum, therefore a nontrivial stratification exists. The same applies to linear hierarchies of three or more nodes, and also to non-linear connected graphs, such as that of observation $\bar{\mathcal{O}}$ of Fig. 2, a stratification of which is proposed in Example 10. However, there are connected graphs for which the finest stratification is the trivial one, such as that in Fig. 4. In case (ii), the observation consists of several DAGs. Each node N in one of such DAGs needs to belong to the same stratum as whichever node belonging to a distinct DAG, since all the nodes of the other DAGs are unrelated to it. This means that $N \diamondsuit N'$ for whichever pair of nodes N and N' that belong to distinct DAGs of the same disconnected observation. Therefore, all the nodes in $\mathcal{N}$ belong to the same equivalence class, i.e. to the same stratum. □

Example 13. Let $\mu''(\bar{\Sigma}) = (\Sigma_0, \bar{\mathcal{V}}, \tilde{\mathcal{O}}^*, \bar{\mathcal{R}})$ be a monitoring problem, where $\tilde{\mathcal{O}}^* = \langle\{N_1\}, \{N_2, N_3\}\rangle$ is a fragmentation of the disconnected observation displayed in Fig. 5. It can be verified that $\Delta(\mu''(\bar{\Sigma})) = \langle\Delta_0'', \Delta_1'', \Delta_2''\rangle$, where $\Delta_0'' = \{\emptyset\}$, $\Delta_1'' = \{\emptyset, \{w\}, \{x\}, \{y\}\}$, $\Delta_2'' = \{\emptyset, \{z\}\}$, therefore the monitoring problem is monotonic although the observation is not stratified. This, however, does not conflict with Proposition 6 since, according to Proposition 4, a stratified observation guarantees monotonicity *independently of the specific system at hand*. In other words, a stratified observation is a sufficient condition for monotonicity, not a necessary one.

A final question is about the nature of the solution of a monitoring problem involving a stratified observation: *What is the relation between Δ_i and Δ_{i+1}?* Proposition 7 gives the answer.

Proposition 7. *Let Δ_i and Δ_{i+1}, $i \in [0\,..\,(n-1)]$, be two consecutive elements in the solution of a stratified monitoring problem. Then,*

$$\forall \delta' \in \Delta_{i+1}\ (\delta' \supseteq \delta, \delta \in \Delta_i). \tag{10}$$

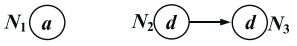


Fig. 5. Disconnected observation for $\bar{\Sigma}$

Proof. This is a Corollary of Proposition 4. Actually, Condition (10) is a definition of monotonic monitoring alternative with respect to Condition (8). In fact, (8) constrains the growth of the actual diagnosis to be monotonic. However, the actual diagnosis is unknown, and variable n in (8) could be assigned any non negative integer value (that is, we do not know when the observation is complete). Therefore, Condition (8) has to hold $\forall \delta = \delta'$, $\delta' \in \Delta_{i+1}$, $i \geq 0$, thus obtaining Condition (10). □

In other words, the set of candidate diagnoses in Δ_{i+1} is obtained by extending a subset of the candidate diagnoses in Δ_i. This is a *shrink and expand* operation, where Δ_i is first shrunk and then the remaining candidates are possibly extended with additional faults, to eventually make up Δ_{i+1}.

Let Γ_{i+1} be the graph including all and only the histories entailing Δ_{i+1}. We can dually say that $\mu(\Sigma)$ is monotonic iff $\forall i \in [0\,..\,(n-1)]$, $\forall h_{i+1} \in \Gamma_{i+1}$, $\exists h_i \in \Gamma_i$ such that h_i is a prefix of h_{i+1}. In summary, monotonicity of candidate diagnoses is a consequence of the monotonic growing of candidate histories, which, in turn, is a consequence of the monotonic growing of candidate traces.

5 Conclusions

This paper deals with monitoring-based diagnosis of DESs under uncertain temporal observations. However, it neither proposes any new method in order to carry out the task nor is focused on the notion of an uncertain observation. Its perspective is rather a conceptual one, so as to span several approaches to the task. Its main points can be summarized as follows:

- The monotonicity property of the solutions (histories and/or fault sets) of a monitoring-based diagnosis problem has been defined, which is important from the point of view of the dependability of such results.
- The lack of monotonicity depends on the incompleteness and temporal uncertainty of the observation.
- Sound and complete monitoring-results are monotonic if the temporal observation is stratified (sufficient condition).

Two existing model-based approaches to DES monitoring with uncertain temporal observations are not monotonic. The former [7] adopts a sound and complete method, but processes a single (logically and temporally uncertain) observable event (and not an observation stratum) at each monitoring step. In the latter [10], the system observation taken into account at each monitoring step consists of a sequence of logically-certain observable events for each component, plus a set of temporal constraints that lead to a partial order of the observable events generated by each considered (sub)system. Therefore the observation is temporally uncertain. The unsolved problem of this approach is

how to frame *temporal windows* in the general case, that is, how to fragment the incoming observation over time in order to achieve completeness of diagnostic results, which is a prerequisite of monotonicity.

In [4], the slicing of an uncertain observation represented as an automaton is dealt with. Such a slicing is *correct* if it guarantees the completeness of the set of candidate histories when the task of a-posteriori diagnosis is performed by either a modular or an incremental method. If the incremental approach is applied also on-line, as suggested by the authors, results are monotonic (although the paper never discusses monotonicity).

A final remark as to the concept of a stratified observation is worthwhile. If the observation taken as input by a sound and complete a-posteriori diagnosis method is stratified, several monotonic intermediate results can be produced. If an intermediate singleton is produced (a unique candidate diagnosis), this means that the faults it includes are certain (in other words, such a diagnosis is a subset of the actual one). More generally, the faults shared by all the candidate diagnoses produced at any intermediate time are included in the actual diagnosis. This piece of information is quite interesting for performing domain-dependent reasoning and/or making decisions (such as, reconfiguration and control) before the diagnostic process is over. Therefore, an intent for the future is to investigate the relationships between diagnosability of DESs and monotonicity of results.

References

1. Baroni, P., Lamperti, G., Pogliano, P., Zanella, M.: Diagnosis of Large Active Systems. Artificial Intelligence 110(1), 135–183 (1999)
2. Console, L., Picardi, C., Ribaudo, M.: Process Algebras for Systems Diagnosis. Artificial Intelligence 142(1), 19–51 (2002)
3. Debouk, R., Lafortune, S., Teneketzis, D.: Coordinated Decentralized Protocols for Failure Diagnosis of Discrete-Event Systems. Journal of Discrete Event Dynamic Systems: Theory and Application 10, 33–86 (2000)
4. Grastien, A., Cordier, M.O., Largouët, C.: Incremental Diagnosis of Discrete-Event Systems. In: 16th International Workshop on Principles of Diagnosis – DX 2005, Monterey, CA, pp. 119–124 (2005)
5. Lamperti, G., Zanella, M.: Diagnosis of Discrete-Event Systems from Uncertain Temporal Observations. Artificial Intelligence 137(1-2), 91–163 (2002)
6. Lamperti, G., Zanella, M.: A Bridged Diagnostic Method for the Monitoring of Polymorphic Discrete-Event Systems. IEEE Transactions on Systems, Man, and Cybernetics – Part B: Cybernetics 34(5), 2222–2244 (2004)
7. Lamperti, G., Zanella, M.: Monitoring and Diagnosis of Discrete-Event Systems with Uncertain Symptoms. In: 16th International Workshop on Principles of Diagnosis – DX 2005, Monterey, CA, pp. 145–150 (2005)
8. Lamperti, G., Zanella, M.: Flexible Diagnosis of Discrete-Event Systems by Similarity-Based Reasoning Techniques. Artificial Intelligence 170(3), 232–297 (2006)
9. Lunze, J.: Diagnosis of Quantized Systems Based on a Timed Discrete-Event Model. IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans 30(3), 322–335 (2000)
10. Pencolé, Y., Cordier, M.O.: A Formal Framework for the Decentralized Diagnosis of Large Scale Discrete Event Systems and its Application to Telecommunication Networks. Artificial Intelligence 164, 121–170 (2005)

11. Rozé, L., Cordier, M.O.: Diagnosing Discrete-Event Systems: Extending the 'Diagnoser Approach' to Deal with Telecommunication Networks. Journal of Discrete Event Dynamic Systems: Theory and Application 12, 43–81 (2002)
12. Sampath, M., Lafortune, S., Teneketzis, D.C.: Active Diagnosis of Discrete-Event Systems. IEEE Transactions on Automatic Control 43(7), 908–929 (1998)
13. Sampath, M., Sengupta, R., Lafortune, S., Sinnamohideen, K., Teneketzis, D.C.: Failure Diagnosis Using Discrete-Event Models. IEEE Transactions on Control Systems Technology 4(2), 105–124 (1996)
14. Su, R., Wonham, W.M.: Global and Local Consistencies in Distributed Fault Diagnosis for Discrete-Event Systems. IEEE Transactions on Automatic Control 50(12), 1923–1935 (2005)

A Service Composition Framework for Decision Making under Uncertainty

Malak Al-Nory[1], Alexander Brodsky[1,2], and Hadon Nash[3]

[1] George Mason University, Virginia, U.S.A.
[2] Adaptive Decisions, Inc., Maryland, U.S.A.
[3] Google Inc., California, U.S.A.
{malnory,brodsky}@gmu.edu, hadonn@gmail.com

Abstract. Proposed and developed is a service composition framework for decision-making under uncertainty, which is applicable to stochastic optimization of supply chains. Also developed is a library of modeling components which include Scenario, Random Environment, and Stochastic Service. Service models are classes in the Java programming language extended with decision variables, assertions, and business objective constructs. The constructor of a stochastic service formulates a recourse stochastic program and finds the optimal instantiation of real values into the service initial and corrective decision variables leading to the optimal business objective. The optimization is not done by repeated simulation runs, but rather by automatic compilation of the simulation model in Java into a mathematical programming model in AMPL and solving it using an external solver.

Keywords: Modelling for stochastic programming, Object-oriented simulation, Supply chain optimization, Decision Support Systems.

1 Introduction

Decision support information systems and frameworks often employ simulation and/or optimization techniques to help decision makers to analyze complex problems and establish actionable recommendations. For example, simulation and optimization have been widely used to minimize costs or maximize profitability in diverse enterprises. Mathematical Programming (MP) and Linear programming (LP) in particular, have been commonly used to model wide range of supply chain optimization problems, such as production and inventory planning, blending products, and network routing. MP tools require constructing a mathematical programming model with decision variables, constraints, and an objective function, possibly using a modelling language such as AMPL [1] or GAMS [2]. Such models can be solved efficiently using existing MP solvers with well-established optimization algorithms, e.g., for Mixed Integer Linear Programming (MILP). Deterministic LP models require complete information and can not solve problems in situations where some data or parameters used in the objective function or the constraints is not available at the time of solving the problem. However, most of supply chain real-world problems, such as

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 363–375, 2009.
© Springer-Verlag Berlin Heidelberg 2009

demand at various nodes in transportation, e.g., [3], and manufacturing networks, e.g., [4], involve making decisions for future events. The increasing complexity of supply chain problems and the variability in the underlying uncertainty models have signified the need for techniques for decision-making under uncertainty.

Stochastic programming techniques [5] are most suitable for supply chain models. The uncertainty can be incorporated in deterministic models by means of random variables to model the uncertain parameters with known probability distributions which make the models considerably more complex. Fortunately, a class of stochastic LP programs can be modeled by their deterministic equivalent, which is an extensive LP formulation of the stochastic program. Clearly, it is easier for modelers to describe the base components of a stochastic program, i.e., the deterministic model and the stochastic process, rather than to define the contingent variables and constraints and build the deterministic equivalent [6].

Building deterministic or stochastic MP models is in general a challenging task, especially for users who are not Operations Research (OR) experts, even for those with general software engineering skills. As indicated in [7], the reason for that challenge is that the elements of a MP model are abstract constraints, which have only an indirect connection to elements of a real-world process. For example, one equation may combine elements from several real-world services or devices. Also, the notions of order and timing of events are usually not explicit in MP models, which puts additional burden on the modeler. Furthermore, the execution of the optimization is typically a black-box for the modeler, with no clear connection to the flow of the real world process. This makes debugging of an optimization model a challenging task. If the optimization fails there is no clear explanation for the failure. Finally, MP models typically lack the modularity of modern object-oriented (OO) programming languages, so they tend to become difficult to maintain over time.

By contrast, stochastic simulations such as discrete-event simulation and Monte Carlo simulation provide means for incorporating complex and stochastic model behavior using easier model development methodologies and tools [8]. Stochastic simulation replicates the behavior of a system by exploiting randomness to obtain statistical sample of possible outcomes. Simulation models are generally well understood by software developers who are typically not OR experts. In addition, simulation offers numerous advantages in ease of modelling, testing, and extensibility. However, simulations are optimized by choosing parameters manually. An optimization layer is usually added by running a simulation multiple times with possible heuristics. It is also possible to combine stochastic simulation and optimization by adding an optimization model on top of a simulation model to optimize a set of user-selected system parameters with respect to some performance measures of the simulation model [9]. The optimization uses the simulation as a black-box and the parameters of the actual problem are not used directly in the optimization model. The improving strategies for the simulation-based approaches are mainly a trial-and-error procedure, thus, simulations lack systematic optimization.

Our approach unifies simulation and stochastic optimization of supply chains, i.e., taking the best of both the simulation and MP optimization worlds. We propose a Stochastic Service Composition (SC)-CoJava framework which allows quick construction of simulation models, with all the advantages of ease of model development, testing and extensibility, while providing MP-based decision optimization. This is achieved by automatic translation of the simulation model into a

stochastic MP model and solving it using an existing MP solver. Stochastic SC-CoJava offers a higher level of abstraction that allows users to model and solve stochastic programs for the supply chain in a relatively fast and easy fashion. Service Composition (SC)-CoJava framework [10, 11] allows specifying a simulation model that is used by SC-CoJava compiler to automatically constructs a MP and then solves it. SC-CoJava also provides a Service Composition framework that allows specifying both atomic and composite services to model supply chain complex problems such as strategic sourcing and transportation. However, SC-CoJava framework provided a modelling environment for deterministic situations only, whereas, most realistic problems involve uncertainty factors that should be incorporated in the model. This is exactly the focus of this paper.

In this paper we propose Stochastic SC-CoJava framework for fast stochastic MP modelling. More specifically, the contributions of this paper are twofold.

First, we developed an extensible modular library of stochastic modelling components, including, Stochastic Service, Scenario, and Random Environment. A Stochastic Service might combine recourse models with resource allocation problems. The service optimizes its decisions now taking into account corrective actions might be taken in future stages. The random parameters are introduced to the model as a finite number of discrete values, and hence, the decision variables can be represented by means of a special structure (i.e., random environment). This library within the Service Composition framework allows quick construction of simulation models for stochastic composite services involving uncertainty parameters.

Second, we developed a case study of Emergency Response stochastic service to exemplify the use of SC-CoJava library of stochastic components and analyzed the results.

There have been work in the literature to allow users to naturally model stochastic programs by extending algebraic modelling languages, e.g., [12], [13], [14], and [15]. These extensions provide data structures and language constructs to support describing the stochastic model. Also, there have been vendor specific extensions and add-on packages to some existing MP languages such as extensions of AMPL [16], and Xpress [17]. These extensions skip the deterministic equivalent model, and provide library of types, functions, and procedures to allow the user to generate the stochastic model components, i.e., the scenario tree and the deterministic equivalent. These extensions require expertise in MP and some are language specific that require familiarity with the base language. Thus, the abstraction level might not be sufficient for the non-OR experts. Model management environments have been also proposed for the algebraic modelling and programming software to allow for integrated development and solution environments, e.g., [18], [19], and [20]. These modelling environments focus on providing options for applying high-level solution algorithms and automatically generating the scenario tree. However, it is not practical to provide the modelers with every possible model. A better approach, which we follow in this paper, is to provide a high-level abstraction in an extensible modelling framework that allows for modelling diverse problems with minimum effort. Perhaps the work that is most related to ours is OptimJ [21] which extends the Java programming language with language support for writing optimization models and abstractions, providing the advantages of modularity, and the flexibility of passing model parameters as Java objects. However, OptimJ models use specialized constructs to represent the optimization model, thus, a simulation model

has to be developed separately in pure Java then linked to the optimization model. OptimJ is essentially an algebraic modeling language in a Java style. In addition, OptimJ does not support easier modelling for stochastic optimization.

This paper is organized as follows. Section 2 explains the problem of stochastic recourse programs. Section 3 describes the simulation semantics of the proposed framework and its individual components. Section 4 defines the optimization semantics. Section 5 exemplifies the use of the framework through a case study. Section 6 describes the implementation architecture. Section 7 concludes the paper and briefly outlines directions for future work.

2 Understanding the Problem

To understand the problem, consider an example of an Emergency Response (ER) supply chain. At the beginning of each time horizon the ER decision makers have to take some initial decisions on how much emergency supply products to purchase from each supplier and how much to store in each warehouse as part of its emergency preparedness plan. When a natural disaster happens, the decision makers needs to take some corrective decisions such as purchasing more products, to deliver to the customers at this time or to store at its warehouses for future use, or moving emergency supply between warehouses to meet demand. At the planning time, we know the suppliers prices at this time, but we are faced with different prices and with a specific demand at the disaster time. From previously collected data, the minimum demand for emergency supply at any disaster time is known. However, we would like to optimize the emergency supply purchases and storage for future too. We can buy additional emergency supply and store it for the disaster time, but storing is associated with certain costs. It may be reasonable to store emergency supply for the disaster time if the price and demand are expected to be high when the disaster takes place and if the storage costs are low. However, we do not know the demand and price at the disaster time in advance.

Problems such as the one in the ER supply chain example are commonly described as a two-stage recourse stochastic programming. Since we have some historical data from previous disasters, it is possible to generate some likely scenarios for the price and demand for the disaster time. Although we do not know in advance which scenario actually takes place, we can still use this information and formulate a two-stage recourse program to optimize the planning decisions for the ER supply chain. In two-stage recourse programs, the decision variables are classified to first-stage variables which are implemented before an outcome of the random variable is realized, while second-stage decision variables model a corrective response to the first-stage decisions and are implemented after an outcome of the random variable is realized. Hence, we make decisions now taking into account that in the future we will be able to take corrective decisions after realizing the random event. A typical two-stage recourse program seeks to minimize the cost of the first-stage decision in addition to the expected cost of the second-stage recourse decision as follows.

$$\begin{aligned} &\textit{min} && f(x) + E_w Q(x,w) \\ &\textit{subject to} && C(x),\ x >= 0 \end{aligned} \tag{1}$$

The nested program (1) minimizes the first-stage cost $f(x)$, and the expectation of multiple programs $Q(x,w)$, each corresponds to a possible scenario for the second-stage, while meeting the first-stage constraints, $C(x)$. Intuitively, variable x denotes the decision on how much emergency supply products ER decision makers should purchase from the suppliers before the disaster happens, and the constraint $C(x)$ denotes the condition that they should at least purchase a minimum amount known from previous data in addition to any capacity constraints. The second-stage is represented by $E_w Q(x,w)$ where $Q(x,w)$ is the following MP.

$$\begin{aligned} &min \quad g(w,y) \\ &subject\ to \quad C'(w,x,y),\ y>=0 \end{aligned} \tag{2}$$

The program (2) minimizes a second-stage cost $g(w,y)$, taking into account the first-stage decision x, and the random event w, subject to the constraint $C'(w,x,y)$. Variable y, is associated with the second-stage cost $g(w,y)$, denotes the decisions on how much products to purchase and to store after the disaster happens and the uncertain demand, denoted by the random variable w, has been realized. The constraint $C'(w,x,y)$ corrects the system after the random even occurred. In our example, this constraint would correspond to satisfying the demand in the disaster time using what we have stored in the planning time in addition to what we would purchase after the disaster happens. Fortunately, the stochastic program above can be expressed in a deterministic form that does not involve randomness. For N scenarios, each occurs with probability p_i, we have

$$\begin{aligned} &min \quad f(x) + \sum_{i=1}^{n} p_i\, g(y_i) \\ &subject\ to \quad C(x),\ C'(x,y_i),\ x>=0,\ y_i>=0\ \ for\ all\ i=1,..n \end{aligned} \tag{3}$$

Again, the program (3) minimizes the first-stage cost $f(x)$, in addition to the expectation of the second-stage costs. Here we introduce a different second-stage y variable for each scenario. The first-stage decision x is feasible for every possible scenario, i.e., x is feasible for both $C(x)$ and $C'(x,y_i)$ for $i = 1,...,N$. Because we solve for all the decisions, x and y_i simultaneously, we are choosing x to be optimal over all the scenarios.

Even though stochastic programs can be modelled by their deterministic equivalent, yet, the indirect connection between the abstract constraints of the model and the elements of real-world process, and the dependencies between these constraints, makes the task of defining these contingent variables and constraints a very challenging one. Stochastic SC-CoJava framework provides a higher level abstraction that allows the modeler to naturally express such constraints in a simulation model using simplified language constructs while providing true decision optimization based on MP.

3 Service Composition Framework

Fig. 1 shows a partially expanded library of supply chain modelling components that adhere to Service Composition (SC)-CoJava framework [10]. The most important

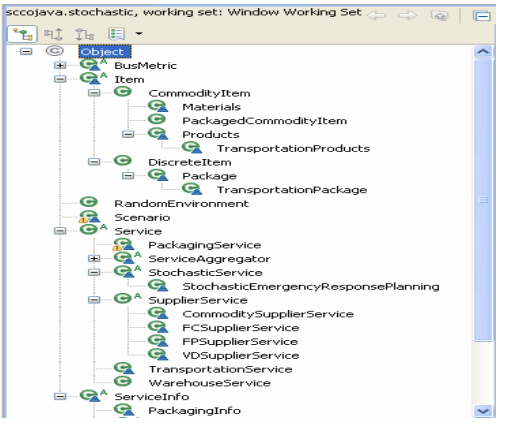


Fig. 1. A snapshot of Stochastic SC-CoJava library

concept is that of a Service, such as Distribution, Manufacturing, and Transportation. Conceptually, a service represents a transformation of incoming Items to outgoing Items. For example, a Manufacturing service transforms Items of type Materials to Items of type Products. Incoming and outgoing Items used in Services are characterized by multiple attributes such as quantity and location, which differ in Items of different types. Services are also associated with one or more Business Metrics, such as Reliability, Responsiveness, and Cost. Also, each Service has an associated Service Information. While Service instance represents a specific dynamic transaction (transformation), its corresponding Service Info instance represents more static parameters. For example, a Supplier Service Info may hold a price list of Items supplied by the Supplier Service, as well as volume discounts and their steps.

Services may be composed of other more basic services such as in our ER example. The ER supply chain can be represented as a complex service which is composed of two sub-services: Commodity Supplier Service, which supplies commodity products; and a Warehouse Service, which stores these products. In turn, each of these sub-services is a complex service composed of more sub-services such as individual suppliers or warehouses. The ER service has only outgoing Items of type Packaged Commodity Item, characterized by quantity, volume, and number of constituting units. Service models are classes in the Java programming language extended with decision variables and assertions. A SC-CoJava user can define a subclass of the Service class and use a special method `Nd.choice(min, max)` to indicate unknown choice constants, i.e., decision variables. The user can also use the *assert(booleanCondition)* construct to indicate Boolean conditions that must be satisfied. SC-CoJava automatically finds an instantiation of real values into every `Nd.choice` in the service object constructor, that satisfies all the *assert* statements and leads to the optimal objective. However, modelling complex services with stochastic parameters and recourse functions, such as our ER service, requires special modelling constructs and a representation for the random events.

Assuming that the underlying stochastic process is discrete and is independent of the state and decision variables of the deterministic model, we extended SC-CoJava framework with the following three classes.

1. `Scenario`. A scenario is used as a data structure to represent an outcome of a random event, e.g., a random demand, or a random delivery or lead times. Typically, recourse programs use scenario trees to show the stages in which random events unfold and decisions are made. The nodes of the event tree represent the state of the discrete state stochastic process at each stage.
2. `RandomEnvironment`. While a `Scenario` is a data structure to represent a single node in the scenario tree, a `RandomEnvironment` is a more complex data structure represented as a class with two members; an array of `Scenario[]` as the observations and any dependent observations, and an array of double probabilities each corresponding to a scenario.
3. `StochasticService`. An abstract class that extends the abstract class `Service` and inherits all the `Service` class data members and methods. A service that extends `StochasticService` automatically builds the recourse problem and solves it. To exemplify how `StochasticService` class is used, consider its subclass `UserStochService` shown below.

```
class UserStochService extends StochasticService{
  /* data members inherited from Service, i.e., serviceInfo,
   * outItem, busMetric, and minFlag */
   RandomEnvironment randomEnv;

  MyStochService(/*constructor data*/){
   /* data instantiation, first-stage Nd.choice and asserts*/
    optObjective();}

  double initialObj(){/*some computations...*/}

  SecondStage secondStage(Scenario s){
           SecondStage ss = new UsersecondStage(s);return ss;}

  //inner class
  class UserSecondStage extends SecondStage{

    //inner class constructor
    UserSecondStage(Scenario s){/*data instantiation,
                      second stage Nd.choice and assertions*/}
    double secondStageObj(){/*some computation...*/}
    }//end of inner class
  }//end of UserStochService class
```

`UserStochService` constructor instantiates the data, such as `randomEnv` and `minFlag`. Also it describes the first-stage decisions and assertions, and at last, and similar to any SC-CoJava `Service`, it invokes `optObjective()` method to optimize the `Service` using the user provided data. `UserStochService` also has `initialObj()` method which computes the initial objective, i.e., the first-stage objective using first-stage decisions defined by in the constructor. The class also implements the method `secondStage(Scenario s)`. This method returns a new `SecondStage` object based on the `Scenario` passed in the method argument.

The inner class `UserSecondStag` extends the abstract class `SecondStage`. The constructor of `UserSecondStag` defines the second-stage decisions and assertions as if `Scenario s` has actually taken place. `UserSecondStage` class provides the method `secondStageObj()` which computes the corrective objective, i.e., a second-stage objective of a corrective decision taken, assuming the realization of the `Scenario s`. Once `optObjective()` method is invoked, the recourse problem is built and the objective of the service decisions (i.e., its business metric objective) is optimized.

4 Optimization Semantics

Assuming that we are given the following:

- a user-defined subclass `UserStochService` of the class `StochasticService`,
- a specific `UserStochService` constructor with `Nd.choice` and *assert* command, so that the same sequence of n `Nd.choice(a`$_i$`,b`$_i$`)` commands is invoked for any input of the constructor.
- a user-defined subclass `UserScenario` of the class `Scenario`.
- an input to the constructor of `UserStochService`, including an instance of `RandomEnvironment` class with scenario objects in `UserScenario`.

Intuitively, `UserStochService` finds the optimal decision choice instantiations that minimize/maximize the objective of `UserStochService`. These optimal instantiations are optimal over all possible scenarios defined by the given instance of `RandomEnvironment`. More formally, we define the semantics of Stochastic SC-CoJava by defining the functions *f(x)*, *C(x)*, *g(w,y)*, *C'(w,x,y)* in the formulation of the two-stage stochastic problem (defined in Section 2).

$$\begin{aligned} &\textit{min}\ \ f(x) + E_w Q(x,w) \qquad (4)\\ &\textit{subject to}\quad C(x),\ x >= 0, \qquad \textit{where } Q(x,w) = \ \textit{min}\quad g(w,y)\\ &\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\quad \textit{subject to}\ \ C'(w,x,y),\ y >= 0 \end{aligned}$$

A *candidate* object of `UserStochService` class is a Java object constructed by the constructor, where real constants $(c_1,\ldots,c_n)$ are used in the sequence of invocation of `Nd.choice(a`$_i$`,b`$_i$`)`, $1 \le i \le n$, command in the constructor. We denote by *CS* the set of all *candidate* objects.

Clearly, every *candidate* object $s \in CS$ is associated with a unique vector $(c_1,\ldots,c_n) \in R^n$; and every vector $(c_1,\ldots,c_n) \in R^n$ is associated with the corresponding *candidate* object *s*, which we will denote as $s(c_1,\ldots,c_n)$. We call a *candidate* object $s(c_1,\ldots,c_n)$ for $(c_1,\ldots,c_n) \in R^n$ *feasible*, if (1) $a_i \le c_i \le b_i$, for every $1 \le i \le n$, where (a_i,b_i) is returned by `Nd.choice(a`$_i$`,b`$_i$`)` and, (2) all *assert* statements in the `UserStochService` constructor are satisfied.

We denote by *FS* the set of all *feasible* `UserStochService` objects, and by *S* the set of corresponding vectors $(c_1,\ldots,c_n)$. Then, the constraint *C* in the stochastic problem is defined as $C(x) \stackrel{def}{=} (x \in S)$.

We now define the function $f: R^n \rightarrow R$ as follows. $f(x) = f(x_1,...,x_n)$ is the value returned by the method `initialObj()`, when applied to the `UserStochService` object $s(x_1,...,x_n)$. For every scenario $w_i \in \{w_1,...,w_k\}$ in the array in `RandomEnvironment` instance, we say that an object of the class `UserSecondStage` is an *candidate* object of that class if it is constructed by its constructor on input Scenario w_i, under the assumption that m invocations of `Nd.choice(l1,u1),..,Nd.choice(lm,um)` return real numbers $(d_1,..,d_m)$ in this order. We denote such an object by $ssi(d_1,..,d_m)$. We say that a *candidate* object $ssi(d_1,..,d_m)$ is *feasible* if (1) for every $1 \leq i \leq m$, $l_i <= d_i <= u_i$, and (2) all *assert* statements in the constructor are satisfied. We denote by *FSSi* the set of all feasible objects $ssi(d_1,..,d_m)$ for scenario w_i, and by *SSi* the set of all corresponding vectors $(d_1,..,d_m) \in R^m$. The constraint $C'(w_i,x,y) = C'(w_i,(x_1,...,x_n),(y_1,...,y_m))$ is then captured by $(y_1,...,y_m) \in SSi$.

We define the function $g:\{1,...,k\} \times R^n \times R^m \rightarrow R$ as follows.

$g(i,(x_1,...,x_n),(y_1,...,y_m))$ is the real value computed by the method `secondStageObj()` applied on `UserSecondStage` object $ssi(y_1,...,y_m)$, which is object $s(x_1,...,x_n)$ constructed given `UserStochService` with input scenario w_i.

The semantics of the `UserStochService` constructor is then defined as follows. It returns an object $s(c_1,...,c_n)$, where $s(c_1,...,c_n)$ is the solution to the following stochastic problem, for the case that `minFlag`=true;

$$min\ f(x_1,...,x_n) + E_i Q(i, x_1,...,x_n, w)\ s.t. (x_1,...,x_n) \in S$$

where $Q(i, x_1,...,x_n) = min\ g(i,(x_1,...,x_n),(y_1,...,y_m))\ s.t.\ (y_1,...,y_m) \in SSi$

Similarly, If `minFlag`=False, the *min* in the problem above should be replaced with *max*. Finally, the semantics of Stochastic SC-CoJava framework is identical to that of the Java language, with the exception of the `UserStochService` constructor, defined in this section.

5 A Case Study

In this section, we exemplify the use and the semantics of the proposed framework using our ER example. Once we have a library of services including `StochasticService`, building a new stochastic service with recourse decisions such as the ER service can be easily done as follows. The user defines a service class `StochasticER` to extend `StochasticServic` similar to `UserStochService` class described in Section 3. A `StochasticER` object can be instantiated by passing the appropriate parameters to the class constructor. `StochasticER` constructor does a number of instantiations using its parameter arguments and uses `Nd.choice(min,max)` to define the first-stage decision variables and the constraints using Java *assert* statements. For example,

```
double initalBuy= Nd.choice(0,maxPurchase);
assert((initialStore+emrItem.getIQty())==iniitalBuy);
```

The implementation of `initialObj()` method only computes the first-stage objective. It instantiates the sub-services, which constitutes the costs of this stage. For example,

```
CommoditySupplierSerAgg SSA= new CommoditySupplierSerAgg(
emrInfo.supInfo, new PackagedCommodityItem(emrItem.iID,
emrItem.iDes,iniitalBuy,emrItem.volume,emrItem.noOfUnits));
```

And the implementation of `secondStage (Scenario s)` method returns a `SecondStage` object as follows.

```
SecondStage secondStage(Scenario s){
  SecondStage ss = new StochSecondStage (s); return ss;}
```

The constructor of the inner (local) class `StochSecondStage`, which extends `SecondStage`, defines decision choices and *asserts* that describe second-stage decision variables and the constraints. The inner class then implements the method `secondStageObj()` which describes the second-stage objective in a similar fashion to `initialObj()` method. Then the sub-services can be easily instantiated using the second-stage decisions.

In the main program we instantiated `StochasticER` using two suppliers' `serviceInfo`, and three warehouses' `serviceInfo`. We also instantiated the demand as shown in the case study data in Table 1.

Table 1. Data set for StochasticER case study

Before-the-disaster data			After-the-disaster scenarios
Demand	Quantity	Volume	Demand
emrItem	150	120	Uniform distribution
Suppliers	Fixed cost	Item price	{150, 200, 250, 300, 330}
S1	200	5.0	
S2	100	7.0	Price
Warehouses	Capacity	Item cost	Uniform distribution
W1	100	0.07	{7.0, 8.0, 9.0, 10.0, 11.0}
W2	150	0.09	
W3	120	0.08	

Every time this application is run as a regular Java program (and the constructor of the `StochasticER` class is invoked), it produces a simulation to the `StochasticER` service example. Because a random selection is returned by every `Nd.choice` method to select such values as quantities of each product to be supplied by each supplier, and quantities stored at each warehouse, each simulation run would result in a different outcome, including a different total cost of `StochasticER` service. These outcomes would not give the minimum cost for the service, but the costs corresponding to the random selections of values in the simulation runs. However, in the optimization semantics, the constructor of `StochasticER` class instantiates an optimal object where the choice values returned by each `Nd.choice` are within the

specified range (min,max), satisfy all the *assert* statements within the scope of these choices, and minimize the objective function. We used ILOG CPLEX (MILP) solver on a Dell OPTIPLEX GX260 machine with Intel® Pentium® 4 CPU 2.80GHz and 1 GB of RAM. It took about 15 seconds to solve the problem for the optimization semantics and produce the optimal object of the StochasticER service with an objective of 2157.00. The optimal object purchases 350 units (from Supplier1) at the planning time, delivers 150 units to customers, and stores 200 units (100 at Warehouse1, and 100 at Warehouse3).

6 Implementation Notes

Stochastic SC-CoJava framework was implemented by extending SC-CoJava [10] with the stochastic modelling components and the StochasticService abstract class shown below.

```
abstract class StochasticService extends Service{
  RandomEnvironment randomEnv;
  abstract double initObj();
  double instantiateProblem(){
       double objective = initObj();
       for(int i=0;i<randomEnv.obs.length; i+=1){
              Scenario s=randomEnv.obs[i];
              SecondStage ss = secondStage(s);
              objective += randomEnv.prob[i]*ss.secondStageObj();
              }
       return objective;
       }
  abstract SecondStage secondStage(Scenario s);
  public double optMetric(){return instantiateProblem();}
  abstract class  SecondStage{
     Scenario s;
     abstract double secondStageObj();
      }//end of SecondStage class
   }//end of StochasticService class
```

StochasticService has two abstract methods, two implemented methods, and an inner (local) abstract class. StochasticService implements the method optMetric() inherited from its super class Service. It simply returns the value of instantiateProblem() method. instantiateProblem() method is also implemented by StochasticService. It first invokes initialObj() and then it defines a Scenario s and a SecondStage ss. Then, it computes the expectation of to the second-stage objective using the value returned by the method secondStageObj() of the object ss. The implementation of secondStage(Scenario s) is merely the instantiation of new SecondStage object using Scenario s. The inner class SecondStage has one abstract method should be implemented by the user secondStageObj(). The procedure defined by instantiateProblem() is a direct translation of the deterministic equivalent of the recourse program into SC-CoJava code. The result is a SC-CoJava simulation procedure that leaves the decision

variables open which we call non-deterministic procedure. Then, the SC-CoJava (and CoJava) compiler translates a non-deterministic simulation procedure into an equivalent decision problem using a reduction algorithm. (See [7] for more details). The resulting decision problem consists of a set of constraints in the modelling language AMPL.

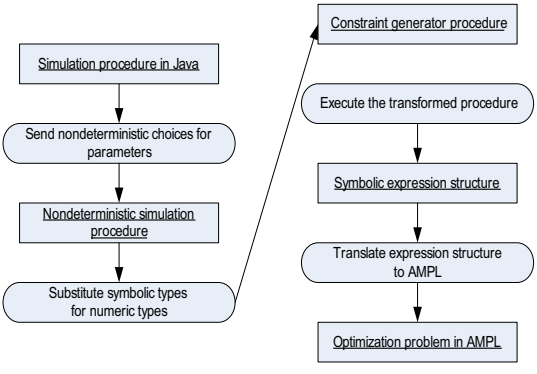


Fig. 2. Overall flow of Implementation Architecture

As shown in Fig. 2, first, a simulation procedure is made non-deterministic by initializing it with values from the non-deterministic choice library, and designating its output as an objective value. This requires no change to the procedure itself, only to its parameters and return value. Next, the procedure is transformed to create a constraint generator procedure. This involves uniformly converting all of its numeric data types to symbolic expression data types. Next, the constraint generator is compiled and executed (using a standard java compiler). The result generated by this procedure is a set of symbolic expression data structures, represent the non-deterministic output of the simulation procedure. These symbolic expressions are translated into a mathematical programming language AMPL and are solved on a solver. Finally, the optimization results are used to run the simulation model deterministically.

7 Conclusions and Future Work

We proposed a new framework and a component library for making decisions under uncertainty. Our approach allows quick construction of composite service with recourse stochastic models with all the advantages of simulation model development, testing and extensibility, and yet allows stochastic optimization based on mathematical programming. Many questions remain for future research. They include extending Stochastic SC-CoJava with approximation techniques, and utilizing the structure of high-level model to develop optimization heuristics that can work in conjunction with existing MP solvers.

References

1. Fourer, R., Gay, D.M., Kernighan, B.W.: AMPL: A Modeling Language For Mathematical Programming. Brooks/Cole-Thomson Learning, Pacific Grove, VA (2003)
2. Brooke, A., Kendrick, D., Meeraus, A., Raman, R.: GAMS: A User's Guide. GAMS Development Corporation (1998)
3. Birge, J.R., Ho, J.K.: Optimal Flows in Stochastic Dynamic Networks with Congestion. Operations Research 41, 203–216 (1992)
4. Lucas, C., Messina, E., Mitra, G.: Risk and Return Analysis of a Multi-period Strategic Planning Problem. In: Thomas, L., Christers, A. (eds.) Stochastic Modelling in Innovative Manufacturing, pp. 81–96. Springer, Berlin (1997)
5. Birge, J.R., Louveaux, F.: Introduction to Stochastic Programming. Springer, New York (1997)
6. Delft, C.v., Vial, J.-P.: A Practical Implementation of Stochastic Programming: An Application to the Evaluation of Option Contracts in Supply Chains. Automatica 40, 743–756 (2004)
7. Brodsky, A., Nash, H.: CoJava: Optimization Modeling by Nondeterministic Simulation. In: van Beek, P. (ed.) CP 2005. LNCS, vol. 3709, p. 877. Springer, Heidelberg (2005)
8. Law, A.M.: Simulation Modeling & Analysis. Suzanne Jeans, New York (2007)
9. Fu, M.C.: Optimization for Simulation: Theory vs. Practice. Informs J. on Comp. 14, 192–215 (2002)
10. Brodsky, A., Al-Nory, M., Nash, H.: Service Composition Language to Unify Simulation and Optimization of Supply Chains. In: 41st Hawaii International Conference on System Sciences. IEEE Press, Hawaii (2008)
11. Al-Nory, M., Brodsky, A.: Unifying Simulation and Optimization of Strategic Sourcing and Transportation. In: Mason, S.J., HIll, R., Moench, L., Rose, O. (eds.) Winter Simulation Conference, Miami, FL (2008)
12. Domenica, N.D., Birbilis, G., Mitra, G., Valente, P.: Stochastic Programming and Scenario Generation within a Simulation Framework: An Information System Perspective. SPEPS 15 (2004)
13. Valente, P., Mitra, G., Poojari, C.A., Kyriakis, T.: Software tools for stochastic programming:a stochastic programming integrated environment (SPInE). Department of Mathematical Sciences, Brunel University, West London, UK (2001)
14. Karabuk, S.: Extending Algebraic Modeling Languages to Support Algorithm Development for Solving Stochastic Programming Models. IMA J. Manag. Math., 1–21 (2007)
15. Entriken, R.: Language Constructs for Modeling Stochastic Linear Programs. Ann. Oper. Res. 104, 49–66 (2001)
16. Fourer, R., Gay, D.M., Kernighan, B.W.: Design Principles and New Developments in the AMPL Modeling Language. In: Kallrath, J. (ed.) Modeling Languages in Mathematical Optimization. Kluwer Academic Publishers, Dordrecht (2003)
17. Dormer, A., Vazacopoulos, A., Verma, N., Tipi, H.: Modeling and Solving Stochastic Problems in Supply Chain Management using Xpress-SP. In: Supply Chain Optimization, pp. 307–354. Springer, US (2005)
18. Dempster, M.A., Scott, J.E., Thompson, G.W.P.: Stochastic Modelling and Optimization using STOCHASTICS. In: Wallace, S.W., Ziemba, W.T. (eds.) Applications of Stochastic Programming, pp. 137–157. SIAM, Philadelphia (2005)
19. Gassmann, H.I., Gay, D.M.: An Integrated Modeling Environment for Stochastic Programming. In: Wallace, S.W., Ziemba, W.T. (eds.) Applications of Stochastic Programming, pp. 159–175. SIAM, Philadelphia (2005)
20. Messina, E., Mitra, G.: Modelling and Analysis of Multistage Stochastic Programming Problems: A Software Environment. Eur. J. Oper. Res. 101, 343–359 (1997)
21. Ateji: `http://www.ateji.com`

A Multi-criteria Resource Selection Method for Software Projects Using Fuzzy Logic

Daniel Antonio Callegari and Ricardo Melo Bastos

Fac. Informática, PUC-RS, Av. Ipiranga 6681, Porto Alegre, Brazil
{daniel.callegari,bastos}@pucrs.br

Abstract. When planning a software project, we must assign resources to tasks. Resource selection is a fundamental step to resource allocation since we first need to find the most suitable candidates for each task before deciding who will actually perform them. In order to rank available resources, we have to evaluate their skills and define the corresponding selection criteria for the tasks. While being the choice of many approaches, representing skill levels by means of ordinal scales and defining selection criteria using binary operations imply some limitations. Pure mathematical approaches are difficult to model and suffer from a partial loss in meaning in terms of knowledge representation. Fuzzy Logic, as an extension to classical sets and logic, uses linguistic variables and a continuous range of truth values for decision and set membership. It allows handling inherent uncertainties in this process, while hiding the complexity from the final user. In this paper we show how Fuzzy Logic can be applied to the resource selection problem. A prototype was built to demonstrate and evaluate the results.

Keywords: Resource Selection, Software Project Management, Fuzzy Logic, Knowledge Representation.

1 Introduction

Resource selection is an important element in project management. Once we define the tasks to be performed, we must select resources for the project and find candidates to each task according to some criteria. Resource selection is, therefore, a fundamental step to resource allocation and project planning. Resource allocation, by its turn, involves resource selection, as well as other types of decisions, including desired workload, timing and cost constraints [1], [2].

Many general approaches to resource selection and allocation deal about resources only in a quantitative manner (where all resources are equally capable). In human-centered activities such as software development, though, knowledge about each available resource is mandatory and clearly determines success [3], [4], [5]. Besides, software development is characterized by frequent changes to the project plan. Every time a change affects tasks and resources, another selection of appropriate resources is often necessary in order to best suit the affected tasks with the people that can perform them.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 376–388, 2009.
© Springer-Verlag Berlin Heidelberg 2009

In mid-size to large companies, there may be a considerable number of resources with different skills to choose from a resource pool. Managers typically assign people to tasks based on their own experience, heuristic knowledge, subjective perception and instinct [6]. In this case, decision support tools and methodologies play an important role for managers as they can help analyzing different configurations and reducing the time needed to perform this activity [7].

In the other side, measuring and classifying people's skill levels as well as defining and running selection criteria can be difficult and time consuming tasks. Many approaches do not deal well with this problem when it comes to knowledge representation, i.e. handling information based on subjective evaluation. Besides, organizations often do not find a perfect matching between available resources and project tasks, as they need to balance current workforce among requiring tasks [8].

The classical methodology on knowledge representation employs conventional two-valued logic and it is very inefficient when dealing with uncertainty and vagueness – notably, it misses a way to represent knowledge based on "common sense". As a consequence, it does not provide an adequate model for reasoning on approximate information (instead of on exact data). Fuzzy Logic provides an efficient conceptual base for dealing with the problem of knowledge representation in uncertain and imprecise environments and its importance is due to the fact that most human reasoning forms – and common sense, in particular – are approximate by their nature; and it also can be used as a decision mechanism [9], [10].

As we have pointed out, in the context of software development, projects involve uncertainties and are subject to frequent adjustments as activities change in response to many events. Hence, we expect that a satisfactory solution to the problem should support approximate reasoning and also be systematic in order to help reassigning resources as projects evolve.

In this paper, we present MRES, a multi-criteria resource selection method based on Fuzzy Logic. A prototype was developed to evaluate both the underlying model and the method. We first review some concepts on resource selection and related work (section 2). The model and the details of the method are presented in sections 3 and 4. Section 5 comments on the evaluation of our proposal. Finally, conclusion and suggestions of future work are presented in section 6.

2 Resource Selection and Knowledge Representation

The resource selection process typically involves decisions based on multiple criteria. Choices are often distinct from manager to manager, but the essence of selecting the most appropriate resources for each project and task can be said to have a common ground: the evaluation of the tasks' requirements aligned with the skills of the available resources. The ultimate goal (resource allocation) is to assign resources to tasks even when the most desirable skills are not available in the resource pool [8].

It is worth to note that finding the "best resource" (for instance, the one with the greatest expertise for a given task) is not always optimal. If a task demands "a high skill level for Java" but also "some experience with HTML", then we should address both requirements considering their respective levels of expected knowledge.

This means that a resource who matches exactly these criteria is more suitable to the task than another resource who is an expert in both Java and HTML – in the second case we would be wasting a resource that could be assigned to a more demanding task. Note that this does not mean the second resource is not an option; on the contrary: unlike common approaches, even if the resource is outside the specified criteria, it should be considered, but as a secondary option.

Therefore, we need to determine some measure of suitability (or fitness) of a resource to a task. By ranking candidate resources for the tasks according to this measure, we can facilitate the forthcoming allocation process.

2.1 Specifying Selection Criteria

A simple approach to resource selection could just allow us to inform the set of skills for each human resource and then specify boolean expressions to select among them (e.g. "knows Java AND knows HTML"). A more elaborated approach could offer some kind of ordinal scale to distinguish levels of experience on each skill (together with a set of relational operators), but selection criteria would remain typically boolean (e.g. "JavaSkills >= 4 AND HtmlSkills > 1").

There are two main problems with such approaches. First, in both cases we end up with a binary partition of the resource space: either a resource passes the selection criteria or not – refer to [3] for instance. Second, there is some uncertainty when someone evaluates both the required skill level for a task and the level of experience of the resources in that skill. Even if we consider continuous skill level measures ranging from 0.0 to 1.0, no manager would ask for a "0.73-level or better Java programmer". Besides, the resource space will still be divided in two separate groups: in this example the phrase would translate to something similar to "JavaSkills >= 0.73", and resources with a skill level as near as 0.72 would not be candidates for that particular task.

While using a continuous interval such as [0.0; 1.0] increases the knowledge representation "resolution", it also impairs human interpretation and understanding. Fortunately Fuzzy Logic allows us to use linguistic terms in order to refer to degrees of measurement. As a result, the formalism creates a linguistic abstraction layer for the user, but still handles all the complex math behind the scenes.

Another problem regarding knowledge representation is that often two managers do not necessarily agree when specifying these levels. As a consequence, while the assignment of grades or quantity levels for skills implies a measurable differentiation among the candidates, we still have to deal with inherent semantic gaps between what we want to say and what is possible to represent in those approaches (and this is not the case where probability would help) [11].

Beginning with Zadeh in his seminal paper "Fuzzy Sets" [9], research has shown that in this case the source of imprecision is the absence of sharply defined criteria of class membership rather than the presence of random variables. In other words, such uncertainty arises from the lack of precise knowledge rather than from probabilistic events [11].

This also clarifies why such levels cannot simply represent a direct mapping to a discrete ordered set, such as *1=low*, *2=average*, *3=high* (one could ask what "average" means and at which point a level changes to "high"). Besides, it is possible

that a manager would want to have five or even more levels in this range, while another manager feels more comfortable with only three; therefore a good solution has to be able to perform translations among different scales, so that managers using different scales could communicate.

It is clear by this example that a certain imprecision is desirable in this process. In other words, it would be better if we could say that a task needs resources with "Java programming skills *near* 0.7". More than that, it would be even better if we could specify criteria like we do in natural language (by using words and not numbers) and in such a way that it would embed some level of uncertainty, but still keeping a continuous range underneath (a continuous range is also needed to allow small updates to a resource's level in a given skill, by means of subsequent individual performance evaluations – not covered in this paper).

2.2 Resource Selection Approaches

There are many approaches to resource selection (and allocation) in the literature. Due to space reasons, we will briefly comment on some work related to our research (see Table 1).

Plekhanova [3] [12], for instance, suggests three categories of relevant information for resource selection: individual data (knowledge and skills), application domain – or ideational – data (such as finance, medicine etc.) and relational data (interrelationships between tasks participants, obtained by task dependencies or team structure). The approach is based on a profile theory; however it uses classical set theory and logic. Otero et. al. [8] present a methodology for resource allocation that considers that time resources spend learning for performing a task is a function of the levels of knowledge in other related skills, and it mentions using Fuzzy Logic as a future work in order to assign resources to tasks.

Table 1. Comparison of MRES and some related works

Item	***MRES***	**[3] [12]**	**[6]**	**[8]**	**[11]**	**[13]**
Individual skills based selection	*yes*	yes	no	yes	n.a.	yes
Discrete or Continuous scale	*C*	D	D	C	n.a.	C
Fuzzy-logic based	*yes*	no	no	similar	yes	yes
Presents strong evidence	*yes*	analytically	yes	yes	analytically	analytically
Deals with social (team) or task relationships	*no*	no	no	partially	no	yes
Has a configurable rule base (inference engine)	*yes*	no	no	partially	partially	no
Domain	*SwDev*	SwDev	SwDev	SwDev	SwDev	Workflow
Main objective	*best fit*	best fit	best fit	best fit	task durations	best fit

Shen et al. [13] apply fuzzy numbers to the task assignment problem in a workflow environment where the idea is to fill abstract roles for tasks, while also dealing with team and task relationships in the solution. Ozdamar and Alanya [11], by their turn, use Fuzzy Logic in a project scheduling model, although the goal is to adjust the duration of tasks, and not for performing resource selection.

Acuña and Juristo [14] go beyond computer science and bring validated experimental results gathered with psychology support to define a Capabilities-Oriented Software Process Model. They present 20 general capabilities said to be critical in software development, categorized in intrapersonal, organizational, interpersonal and management skills. The solution, however, is role-oriented (not individually oriented).

All the analyzed works (some of them listed on Table 1) have their domain in software development, except one [13], which relates to workflow systems in general. As we can see, some of them are also based on Fuzzy Logic (though sometimes with distinct objectives), and one presents a similar approach. In addition, related works in Table 1 present evidences for their respective approaches, even though some of them only analytically (they do not present results from a survey or controlled experiment, for instance).

Whereas providing important contributions, most approaches usually do not appropriately handle the problems mentioned in the previous section or they present hard to use complex solutions. In particular, we are interested in addressing the problem in depth, but also hiding the complexity from the final user.

The use of Fuzzy Logic allows us to deal with the inherent uncertainties in this process and also interface with the user in natural language (by using linguistic terms and variables).

In this paper we assume the manager as the role performing resource selection.

2.3 Evolving the Proposal

Our Fuzzy Logic based approach allows the manager to specify selection criteria such as "*high level* of experience with Java and *some knowledge* on HTML". Here, the definitions for "high level of experience" and "some knowledge" embrace that desired imprecision. Also, we do not induce a bi-partition of the resource space anymore, because each resource will belong to the group of candidate resources with more or with less degree. In fuzzy set theory, this degree represents the membership of an element to a set. The more the element is related to the set, the greater its membership degree gets (values are in the normalized range from 0.0 – no membership; to 1.0 – full membership).

Fuzzy sets can then be mapped to linguistic terms (such as *low*, *medium* and *high* – each one allowing some uncertainty, or *fuzziness*) and grouped into fuzzy (linguistic) variables (see Fig. 1 for an example in the domain of temperatures). As the "real" temperature is updated (horizontal axis), the membership for each of the terms (fuzzy sets) varies. Each term represents a linguistic and imprecise concept regarding a level of temperature. In the example, a temperature of 18ºC is considered *low* to a degree of 0.15 and *medium* to a degree of 0.5 (one could describe it as "medium but a little bit low").

Instead of using regular "crisp" values, we can now use fuzzy terms to represent a measure (and include some level of imprecision). The wider is the shape of the term,

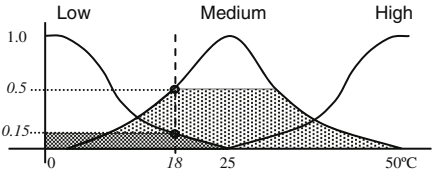


Fig. 1. An example of a fuzzy variable and its fuzzy terms

the more imprecision its value embeds. If a manager wants "more precision", s/he can use a greater number of narrower terms.

As a consequence of the issues we have presented, we addressed the selection problem considering the following:

1. **Knowledge representation:** The solution should represent skill levels for tasks and resources in a continuous scale in order to better handle vagueness (otherwise there would be big leaps from one level to another) and also small changes in the levels due to an afterward performance evaluation process;
2. **Selection criteria:** We must provide an appropriate way to specify the selection criteria in a non-binary form;
3. **Imprecision support:** Ability to handle inherent vagueness and uncertainty;
4. **Scaling:** It must support multiple term sets (e.g. ordinal scales with any number of term) but keep a normalized scale underneath; and
5. **Flexibility:** The solution must be easily configurable since different managers and organizations may want to use different policy and criteria, as well as different set of terms for describing skill levels.

3 MRES – A Fuzzy Logic Based Approach

Our approach to resource selection, called MRES (multi-criteria resource selector), was built with the goal to output a ranked list of the most suitable resources for any given project task, by using a multi-valued logic and a configurable set of inference rules. This configurable rule base is intended to mimic the desired decision policy defined by the manager.

In order to explain the proposed model, we must define some fundamental concepts (we assume the reader is familiar with Fuzzy Logic). The model is composed by (a) a set of tasks, (b) a set of resources, (c) a set of skills, (d) linguistic terms representing possible levels of skills for each skill-task and skill-resource pairs, (e) linguistic terms for levels of suitability, and (f) a set of inference rules that translate the desired selection policy.

As a simple example, consider skills such as knowledge on PHP, SQL and HTML, and a task "Validate login information in home page". Let us assume the linguistic terms "some knowledge", "good knowledge" and "expert", and the following desired mapping for performing the task: {(*PHP*, *expert*), (*SQL*, *good knowledge*), (*HTML*, *some knowledge*)}.

Then, considering resources Anna, John and Mary with the mapping on Table 2, we must rank them in accordance to the task's expected skills and levels (a subsequent allocation process would then select as many resources as needed for each task, balancing their respective workforces).

Table 2. Sample mapping of skills and resources

	Anna	**John**	**Mary**
PHP	some knowledge	good knowledge	expert
SQL	good knowledge	good knowledge	some knowledge
HTML	expert	good knowledge	some knowledge

At the core of MRES there is a Fuzzy Logic based inference system with two input variables and one output variable (Fig. 3). The first input variable represents an expected skill level for a task while the second input variable represents the current level of a resource in that skill. A typical fuzzy inference engine is run several times for combinations of resources and the expected skill levels for a task. For each resource, it computes the level of suitability (the output variable). After that, a final step ranks the available resources according to their respective suitability levels for the particular task. Fig. 2 presents the main elements of the model.

A fuzzy term is defined by a tuple (name, type, x1; x2; x3; x4). We use standard membership functions (S, Z, Pi, Lambda) for the type, which can be easily defined by a set of four points (details follows). The weight of a term is defined by W = (x1+x4)/2, for simplicity [15].

Note that each task expects (as opposed to "requires") a set of skills, which means that the most suitable resources available will be selected even if they do not match all the skills (recall section 2). The process of resource selection in MRES is comprised of the following steps:

1. Input resources, tasks and skill data;
2. Define the fuzzy terms for FVR, FVT, FVS.
3. Map expected skills for each activity (for each expected skill, choose the corresponding term from FVTt);
4. Map each resources' skills (for each skill for a resource, choose the corresponding term from FVRt);
5. Define the set of inference rules;
6. Run selection algorithm for a task *t*.

In step 1 the manager inputs the available resources, the tasks for the project and the set of skills that will be evaluated. In step 2, the manager configures each of the fuzzy variables by defining their respective terms. Each linguistic term is represented by a fuzzy set, defined by four points in (x,y) cartesian coordinates [(x1, 0.0); (x2, 1.0); (x3, 1.0); (x4, 0.0)], in order to make it simpler for the managers to define the terms and also to simplify the computation of membership values. The cartesian coordinates are limited from 0.0 to 1.0 in both axes. The vertical axis corresponds to the membership and the horizontal axis to the skill level.

MRES Model (main elements):

TSK = A set of tasks {t1, t2, t3, …}.
RES = A set of resources {r1, r2, r3, …}.
SKL = A set of skills {s1, s2, s3, …}.
FVT = A fuzzy variable representing the expected level of skill for a task.
FVR = A fuzzy variable representing the current level of a resource for a skill.
FVS = A fuzzy variable representing the suitability of a resource to a task regarding a given skill.
FVTt = A set of fuzzy terms for the variable FVT.
FVRt = A set of fuzzy terms for the variable FVR.
FVSt = A set of fuzzy terms for the variable FVS.
TSM = Task-skill mapping {(t1,s1,*level*), (t1,s7,*level*), (t2,s2,*level*), …}.
RSM = Resource-skill mapping {(r1,s1,*level*), (r1,s2,*level*), (r2,s4,*level*), …}.
RUL = A set of fuzzy rules in the form:
IF FVT=tt AND FVR=lt THEN FVS=st
(two antecedents and one consequent).

Fig. 2. Main elements of the MRES model

The manager can define as many terms as s/he wants for each variable. For example, {*low*, *medium* and *high*} or {*minimum*, *acceptable*, *good*, and *excellent*}. When the manager associates a skill to a task or a resource (steps 3 and 4), s/he chooses a term from FVTt or FVRt, respectively.

When performing this process for the first time, the actual assigned value is the weight of the chosen term (the middle point of the set's base in the x axis). This means that the manager's choice remains in the language domain (for instance, "high") but the model translates it to a crisp value that represents the chosen term in this first assignment.

There are two aspects to observe here. First, this associated value can be changed gradually in time when the task is modified or, more likely, as the resource increases his/her skill level. As mentioned before, this is part of an evaluation process that is beyond the scope of this paper. Second, it allows changing the terms for the respective variable in number and in their definition (shape) at any time, since the actual crisp value remains unchanged. As a consequence, two managers who use different sets of terms can share the same resource and task base.

Let us assume the terms {*zero*, *low*, *med*, *high*} for the suitability variable. Then in step 5 we choose a term from this set for each pair of the cartesian product between the terms in FVRt and FVTt. This creates a matrix of rules such as the one shown in Table 3, assuring that every possibility is covered by a rule.

Considering the evaluation of the suitability for the skill "Java", then the rules can be read such as: "If the task expects *good knowledge* in Java and the resource's current skill level is *good*, then suitability is *high*".

Table 3. A sample fuzzy rule matrix

Task Resource	Some knowledge	**Good knowledge**	Expert
None	*zero*	*zero*	*zero*
Acceptable	*high*	*med*	*low*
Good	*med*	***high***	*med*
Excellent	*low*	*med*	*high*

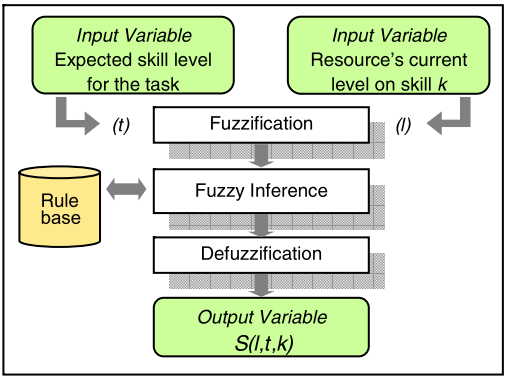


Fig. 3. Determining the suitability for a specific skill

The set of rules represent a policy on resource selection. For example, if the manager wants to allow that people with a higher skill level than the one expected by the task to be ranked as good candidates, than s/he only has to redefine the suitability terms for the lower left part of the matrix. Note that the example in Table 3 delivers the opposite: an "excellent" resource for a task expecting just "some knowledge" for a given skill will not be considered a good match (which translates to a "low" suitability) – perhaps we could save such a skilled resource for a more demanding task and find some junior resource to work on that one.

The suitability of a resource's current level *l* to a task's expected level *t* in a skill *k*, called *S(l,t,k)*, is determined by the standard set of fuzzy steps depicted in Fig. 3. Fuzzification is the process of converting the (input) crisp values to their respective fuzzy values (refer to Fig. 1 again). The inference step evaluates the rules and determines a composition of the terms for the output variable. Finally, the defuzzification step converts the results from the inference step to a crisp value again.

We use a classic centroid defuzzification method (1) for determining the crisp value of the output variable – details can be obtained in [15]:

$$crispvalue = \frac{\sum_{r=1}^{n} W_r \cdot A_r}{\sum_{r=1}^{n} A_r} \quad (1)$$

The elements in the formula are listed below:

- *n* is number of rules in fuzzy system (|RUL|);
- *Wr* is the weight of consequent term in FVSt in fuzzy rule $r \in$ RUL; and
- *Ar* is the area of the consequent term in FVSt for rule *r*, limited to the membership value of the antecedent ("if" part).

Note that the values for *l* and *t* are acquired from a database given the particular resource and the task. The output value is calculated for every expected skill in the

task. Then, the overall resource suitability (2) for the task is determined by the product of all *S(l,t,k)* where n is the number of skills expected by the task:

$$suitability(resource, task) = \prod_{k=1}^{n} S(l,t,k) \tag{2}$$

By computing the suitability using this formula we assure the results will always remain in the normalized interval of 0.0 to 1.0, which is consistent with the goals defined in section 2.3. After this process, the final listing of ranked resources can also be reduced in size by defining a minimum suitability threshold.

4 Prototype

A prototype was built to demonstrate the use of the model in practice. The prototype allows us to easily perform all the steps described in the previous section as well as exchange data with a commercial project management product.

The interface provides screens to define the shape of the linguistic terms, to configure the rule base (Fig. 4a), and to perform simulations with the current values helping to adjust the desired selection policy (Fig. 4b).

The prototype has a "debug" mode, which shows all internal values and calculation logs if needed. It is, however, intended to be used in "normal" mode, where only concepts appear to the user (linguistic terms and variables). Following this idea, the calculated suitability levels for each of the selected resources are translated to plain English according to the terms defined for the suitability variable. If the corresponding crisp value remains inside a single term, then the exact name of the term is used. Otherwise expressions such as "a little *high*" or "between *low* and *medium*" are used.

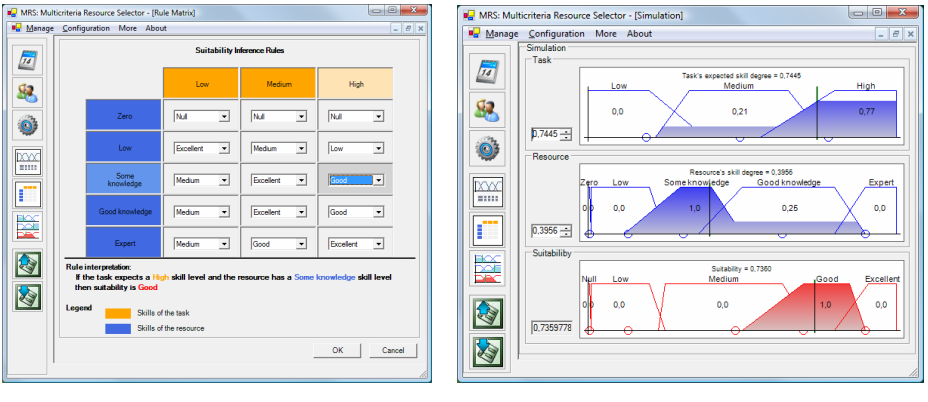


Fig. 4. (a) Matrix of Inference Rules in MRES, and (b) Simulation screen showing sample fuzzy variables in MRES. Small circles represent the weight of the terms.

Also, because all values are stored in a normalized range, even if the manager changes the terms (in number and/or definition), the solution remains robust and the generated natural language phrases will reflect every change accordingly.

5 Evaluation

In order to evaluate our approach, we took the same problem and results from a survey made with 30 participants from both industry and academia and compared with the results from MRES. The survey conducted in [8] asked all participants to manually rank a list of six resources in a sample scenario, based on their capabilities in the required skills for a given task description.

The task demanded expert knowledge in C++, good knowledge of Windows NT and 2000 operating systems and experience with programming a special kind of hardware. Some resources were also skilled in VB and Java programming languages. Skills were classified as *Low*, *Intermediate* and *High*.

By inputting the equivalent information in our prototype we could analyze the results and compare it with the average solution provided by the participants. Fig. 5 presents part of the model for this scenario. Resources are named *eng1* to *eng6*.

Even though we had to define the parameters for the fuzzy part of the model, all the remaining information was used exactly as specified. After running the method, the final ranking of the available resources in MRES was identical to the ranking obtained by the survey participants. This means that the method (and the underlying model) was able to correctly emulate the selection procedure of the survey participants.

Note that the suitability values for *eng3* and *eng2* are the same due to their originally similar levels on the respective skills and the corresponding fuzzy values determined by the shape of the fuzzy terms. Also, the size of the sample and the characteristics of the survey make this result statistically valid.

Finally, we should note that the solution proposed in [8] presents another aspect that was not considered at this time: a similarity function over the skills, so that the mechanism may infer other possibilities when a specific skill is not present for a resource (e.g. task requires knowledge on C++ but resource knows Java). Then they compute the suitability according to the previously specified level of similarity between those skills.

However, by grouping skills by their fundamental concepts (e.g. replacing “C++” and “Java” by “Object Oriented Programming”) we were again able to obtain the very same results as them, besides having the advantages of using Fuzzy Logic. In this sense, this work confirms that solutions following these approaches provide satisfactory results and also it contributes to them by allowing better handling of knowledge representation and uncertainty.

Although not being the goal of this paper, the solution can be extended for selecting people based also on soft skills and covering other types of requirements, such as ability to negotiate, ability to work in teams, leadership, and so on. It can also be used with other types of resources, such as equipment, for example.

Scenario data from [8] and corresponding results in MRES

(details omitted due to space reasons)

TSK = {t1}
RES = {eng1, eng2, eng3, eng4, eng5, eng6}
SKL = {C++, WinNT, Win2000, HW, VB, Java}
FVTt = {(low,Z,0;0;0.2;0.4), (med,Pi,0.3;0.45;0.55;0.7), (high,S,0.6;0.8;1;1)}
FVRt = {(low,Z,0;0;0.2;0.4), (intermediate,Pi,0.3;0.45;0.55;0.7), (high,S,0.6;0.8;1;1)}
FVSt = {(low,Z,0;0;0.2;0.4), (average,Pi,0.3;0.45;0.55;0.7), (high,S,0.6;0.8;1;1)}
TSM = {(t1,C++, high), (t1,WinNT, med), (t1,HW, low), …}
RSM = {(eng1,c++,low), (eng1,WinNT,intermediate), (eng3,HW,high), …, (eng6,VB,low), …}
RUL = {(low,low,high), …, (med,intermediate,high), … (high,high,high)}

* * * * *

Our results (ranked resources and respective suitabilities):

#	*Resource*	*Suitability*
1st :	eng5	0.1600
2nd :	eng3	0.1000
3rd :	eng2	0.1000
4th :	eng6	0.0640
5th :	eng4	0.0400
6th :	eng1	0.0256

Fig. 5. Model evaluation data and results

6 Final Remarks and Future Work

This paper discussed common problems related to resource selection in software projects. In special, knowledge representation and selection criteria definition were addressed in order to help improving current approaches. A method that helps decision support for selecting human resources for software projects based on Fuzzy Logic was introduced and a prototype was built and evaluated.

The use of Fuzzy Logic brings two main advantages for the proposed method: first, it appropriately handles uncertainty; and, second, it isolates the complexity of numbers from the final user. In addition, we showed that MRES properly emulates the decision made by humans for the same circumstances.

As another important contribution, the ability to reconfigure at any time the number and definition of linguistic terms without loosing already acquired and provided data (skill levels for tasks and resources), as well as the suitability rules, allows managers with different visions to share information.

This method is part of a solution under development by the authors in the area of dynamic reconfiguration of software projects, where the final goal is to provide decision support to managers when certain events affect resources and activities in software projects during their execution [16].

During the development of a software product, for instance, a change may affect the required skills of a task. A resource allocation solution has to consider many combinations of resources and tasks, whose inputs are taken from the selection mechanism.

Acknowledgement. Study developed by the Research Group of the PDTI 01/2008, financed by Dell Computers of Brazil Ltd. with resources of Law 8.248/91.

References

1. Schwalbe, K.: Information Technology Project Management, Thomson Learning, 2nd edn., Canada (2002)
2. Kerzner, H.: Applied project management: best practices on implementation. John Wiley & Sons, Chichester (2000)
3. Plekhanova, V.: On Project Management Scheduling where Human Resource is a Critical Variable. In: Gruhn, V. (ed.) EWSPT 1998. LNCS, vol. 1487, pp. 116–121. Springer, Heidelberg (1998)
4. Joslin, D., Poole, W.: Agent-Based Simulation for Software Project Planning. In: Proceedings of the 37th Conference on Winter Simulation, pp. 1059–1066 (2005)
5. Cugola, G., Di Nitto, E., Fuggetta, A., Ghezzi, C.: A framework for formalizing inconsistencies and deviations in human-centered systems. ACM Transactions on Software Engineering 5(3), 191–230 (1996)
6. Acuña, S.T., Juristo, N., Moreno, A.M.: Emphasizing Human Capabilities in Software Development. IEEE Software 23(2), 94–101 (2006)
7. Royce, W.: Software Project Management: A Unified Framework. Addison-Wesley, Reading (1998)
8. Otero, L.D., Centeno, G., Torres, A.R., Otero, C.E.: A Systematic Approach of Resource Allocation in Software Projects. Computers & Industrial Engineering 55, 4 (2008)
9. Zadeh, L.A.: Fuzzy sets. Information and Control 8, 338–353 (1965)
10. Cox, E.D.: Fuzzy Logic for Business and Industry. Charles River Media (1995)
11. Ozdamar, L., Alanya, E.: Uncertainty modelling in software development projects (with case study). Annals of Operations Research 102(6), 157–178 (2001)
12. Plekhanova, V.: Applications of the Profile Theory to Software Engineering and Knowledge Engineering. In: Proceedings of the Twelfth International Conference on Software Engineering and Knowledge Engineering, Knowledge Systems Institute, pp. 133–141 (2000)
13. Shen, M., Tzeng, G., Liu, D.: Multi-Criteria Task Assignment in Workflow Management Systems. In: Proceedings of the 36th Hawaii International Conference on System Sciences. IEEE Press, Los Alamitos (2003)
14. Acuña, S.T., Juristo, N.: Modelling human competencies in the software process. In: ProSim 2003 (2003)
15. Kosko, B.: Neural Networks and Fuzzy Systems: A Dynamical Systems Approach to Machine Intelligence. Prentice-Hall, Englewood Cliffs (1991)
16. Callegari, D.A., Bastos, R.M.: A Systematic Review of Dynamic Reconfiguration of Software Projects. In: SBES 2008 - XXII Simpósio Brasileiro de Engenharia de Software, pp. 299–313 (2008)

An Optimized Hybrid Kohonen Neural Network for Ambiguity Detection in Cluster Analysis Using Simulated Annealing

E. Mohebi and M.N.M. Sap

Faculty of Computer Science and Information System,
University Technology Malaysia 81310 Skudai, Johor, Malaysia
mehsan3@siswa.utm.my, mohdnoor@utm.my

Abstract. One of the popular tools in the exploratory phase of Data mining and Pattern Recognition is the Kohonen Self Organizing Map (SOM). The SOM maps the input space into a 2-dimensional grid and forms clusters. Recently experiments represented that to catch the ambiguity involved in cluster analysis, it is not necessary to have crisp boundaries in some clustering operations. In this paper to overcome the ambiguity involved in cluster analysis, a combination of Rough set Theory and Simulated Annealing is proposed that has been applied on the output grid of SOM. Experiments show that the proposed two-stage algorithm, first using SOM to produce the prototypes then applying rough set and SA in the second stage in order to assign the overlapped data to true clusters they belong to, outperforms the proposed crisp clustering algorithms (i.e. I-SOM) and reduces the errors.

Keywords: Clustering, Ambiguity, Self Organizing Map, Rough set, Simulated Annealing.

1 Introduction

The Self Organizing Map (SOM) proposed by Kohonen [1], has been widely used in industrial applications such as pattern recognition, biological modeling, data compression, signal processing and data mining [2],[3],[4]. It is an unsupervised and nonparametric neural network approach. The success of the SOM algorithm lies in its simplicity that makes it easy to understand, simulate and be used in many applications. The basic SOM consists of neurons usually arranged in a two-dimensional structure such that there are neighborhood relations among the neurons. After completion of training, each neuron is attached to a feature vector of the same dimension as input space. By assigning each input vector to the neuron with nearest feature vectors, the SOM is able to divide the input space into regions (clusters) with common nearest feature vectors. This process can be considered as performing vector quantization (VQ) [5]. In addition, because of the neighborhood relation contributed by the inter-connections among neurons, the SOM exhibits another important property of topology preservation.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 389–401, 2009.
© Springer-Verlag Berlin Heidelberg 2009

Clustering algorithms attempt to organize unlabeled input vectors into clusters such that points within the cluster are more similar to each other than vectors belonging to different clusters [6]. The clustering methods are of five types: hierarchical clustering, partitioning clustering, density-based clustering, grid-based clustering and model-based clustering [7]. The rough set theory employs two upper and lower thresholds in the clustering process, which result in a rough clusters appearance. This technique also could be defined in incremental order i.e. the number of clusters is not predefined by users.

In this paper, a new two-level clustering algorithm is proposed. The idea is that the first level is to train the data by the SOM neural network and the clustering at the second level is a rough set based incremental clustering approach [8], which will be applied on the output of SOM and requires only a single neurons scan. The optimal number of clusters can be found by rough set theory, which groups the given neurons into a set of overlapping clusters (clusters the mapped data respectively). Then the overlapped neurons will be assigned to the true clusters they belong to, by apply simulated annealing algorithm. A simulated annealing algorithm has been adopted to minimize the uncertainty that comes from some clustering operations. In our previous work [3], the hybrid SOM and rough set has been applied to catch the overlapped data only, but the experiment results show that the proposed algorithm (SA-Rough SOM) outperforms the previous one.

This paper is organized as following; in section 2, the basics of SOM algorithm are outlined. The Incremental Clustering and Rough set theory are described in section 3. In section 4, the essence of simulated annealing is described. The proposed algorithm is presented in section 5. Section 6 is dedicated to experiment results, section 7 provides brief conclusion, and future works.

2 Self Organizing Map and Clustering

Competitive learning is an adaptive process in which the neurons in a neural network gradually become sensitive to different input categories, sets of samples in a specific domain of the input space. A division of neural nodes emerges in the network to represent different patterns of the inputs after training.

The division is enforced by competition among the neurons: when an input x arrives, the neuron that is best able to represent it wins the competition and is allowed to learn it even better. If there exist an ordering between the neurons, i.e. the neurons are located on a discrete lattice, the competitive learning algorithm can be generalized. Not only the winning neuron but also its neighboring neurons on the lattice are allowed to learn, the whole effect is that the final map becomes an ordered map in the input space. This is the essence of the SOM algorithm. The SOM consist of m neurons located on a regular low-dimensional grid, usually one or two dimensional. The lattice of the grid is either hexagonal or rectangular.

The basic SOM algorithm is iterative. Each neuron i has a d-dimensional feature vector $w_i = [w_{i1},...,w_{id}]$. At each training step t, a sample data vector $x(t)$ is randomly chosen for the training set. Distance between $x(t)$ and all feature vectors

are computed. The winning neuron, denoted by c, is the neuron with the feature vector closest to $x(t)$:

$$c = \arg\min_i \|x(t) - w_i\|, \qquad i \in \{1,..., m\} \tag{1}$$

A set of neighboring nodes of the winning node is denoted as N_c. We define $h_{ic}(t)$ as the neighborhood kernel function around the winning neuron c at time t. The neighborhood kernel function is a non-increasing function of time and of the distance of neuron i from the winning neuron c. The kernel can be taken as a Gaussian function:

$$h_{ic}(t) = e^{-\frac{\|Pos_i - Pos_c\|^2}{2\sigma(t)^2}} \tag{2}$$

where Pos_i is the coordinates of neuron i on the output grid and $\sigma(t)$ is kernel width. The weight update rule in the sequential SOM algorithm can be written as:

$$w_i(t+1) = \begin{cases} w_i(t) + \varepsilon(t) h_{ic}(t)\big(x(t) - w_i(t)\big) \forall i \in N_c \\ w_i(t) \qquad\qquad ow \end{cases} \tag{3}$$

Both learning rate $\varepsilon(t)$ and neighborhood $\sigma(t)$ decrease monotonically with time. During training, the SOM behaves like a flexible net that fold onto a cloud formed by training data. Because of the neighborhood relations, neighboring neurons are pulled to the same direction, and thus feature vectors of neighboring neurons resemble each other. There are many variants of the SOM [9], [10]. However, these variants are not considered in this paper because the proposed algorithm is based on SOM, but not a new variant of SOM.

3 Incremental Clustering and Rough set Theory

3.1 Incremental Clustering

Incremental clustering [11], is based on the assumption that it is possible to consider data points one at a time and assign them to existing clusters. Thus, a new data item is assigned to a cluster without looking at previously seen patterns. Hence the algorithm scales well with size of data set.

It employs a user-specified threshold and one of the patterns as the starting leader (cluster's leader). At any step, the algorithm assigns the current pattern to the most similar cluster (if the distance between pattern and the cluster's leader is less or equal than threshold) or the pattern itself may get added as a new leader if its similarity with the current set of leaders does not qualify it to get added to any of the existing clusters. The set of leaders found acts as the prototype set representing the clusters

and is used for further decision making. A high-level description of a typical incremental algorithm is as following pseudo code [12].

```
Incremental_Clustering (Data, Thr){
 Cluster_Leader = d1;
 While (there is unlabeled data){
  For (i = 2      to    N)
   If (dis (Cluster_Leader, di) <= Thr)
    Put di in the same cluster as Cluster_Leader;
   Else
     // new Cluster
    Cluster_Leader = di;
 }//end of while
}
```

An incremental clustering algorithm for dynamic information processing was presented by [13]. The quality of a conventional clustering scheme is determined using within group error [14] Δ given by:

$$\Delta = \sum_{i=1}^{m} \sum_{u_h, u_k \in C_i} distance\ (u_h, u_k) \tag{4}$$

u_h, u_k *are objects in the same cluster* C_i.

3.2 Combination of Rough Set Theory and Incremental Clustering

This algorithm is a soft clustering method employing rough set theory [15]. It groups the given data set into a set of overlapping clusters. Each cluster is represented by a lower approximation and an upper approximation $(\underline{A}(C), \overline{A}(C))$ for every cluster $C \subseteq U$. Here U is a set of all objects under exploration. However, the lower and upper approximations of $C_i \in U$ are required to follow some of the basic rough set properties such as:

1. $\emptyset \subseteq \underline{A}(C_i) \subseteq \overline{A}(C_i) \subseteq U$
2. $\underline{A}(C_i) \cap \underline{A}(C_j) = \emptyset,\ i \neq j$
3. $\underline{A}(C_i) \cap \overline{A}(C_j) = \emptyset,\ i \neq j$
4. If an object $u_k \in U$ is not part of any lower approximation, then it must belong to two or more upper approximations.

Note that (1)-(4) are not independent. However enumerating them will be helpful in understanding the basic of rough set theory.

The lower approximation $\underline{A}(C)$ contains all the patterns that definitely belong to the cluster C and the upper approximation $\overline{A}(C)$ permits overlap. Since the upper approximation permits overlaps, each set of data points that are shared by a group of clusters define indiscernible set. Thus, the ambiguity in assigning a pattern to a cluster

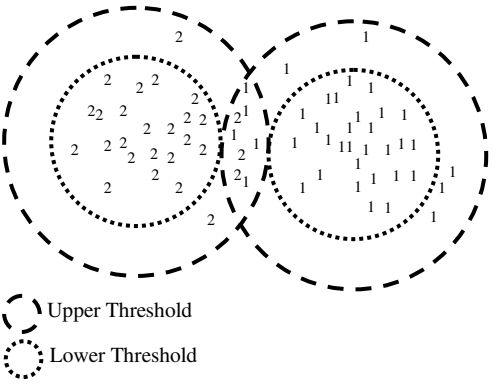


Fig. 1. Rough set Incremental Clustering. Upper and lower approximations for two clusters are depicted.

is captured using the upper approximation. Employing rough set theory, the proposed clustering scheme generates soft clusters (clusters with permitted overlap in upper approximation) see Fig. 1. A high-level description of a rough incremental algorithm is as following pseudo code [16].

```
Rough_Incremental (Data, upper_Thr, lower_Thr){
Cluster_Leader = d1;
 While (there is unlabeled data){
  For (i = 2      to    N)
   If (distance(Cluster_Leader, di) <= lower_Thr)
      Put di in the lower approx of Cluster_Leader;
   Else If (distance(Cluster_Leader, di) <= upper_Thr)
      Put di in all existing clusters (j=1 to k)that
      distance(Cluster_Leaderj, di) <= upper_Thr ;
   Else
      // new Cluster
      Cluster_Leader = di;
 }//end of while
}
```

For a rough set clustering scheme and given two objects $u_h, u_k \in U$ we have three distinct possibilities:

1. Both u_k and u_h are in the same lower approximation $\underline{A}(C)$.
2. Object u_k is in lower approximation $\underline{A}(C)$ and u_h is in the corresponding upper approximation $\overline{A}(C)$, and case 1 is not applicable.
3. Both u_k and u_h are in the same upper approximation $\overline{A}(C)$, and case 1 and 2 are not applicable.

For these possibilities, three types of equation (4) could be defined as following:

$$\Delta_1 = \sum_{i=1}^{m} \sum_{u_h, u_k \in \underline{A}(X_i)} distance\ (u_h, u_k)$$
$$\Delta_2 = \sum_{i=1}^{m} \sum_{u_h \in \underline{A}(X_i)\ and\ u_k \in \overline{A}(X_i)} distance\ (u_h, u_k) \tag{5}$$
$$\Delta_3 = \sum_{i=1}^{m} \sum_{u_h, u_k \in \overline{A}(X_i)} distance\ (u_h, u_k)$$

The total error of rough set clustering will then be a weighted sum of these errors:

$$\Delta_{total} = w_1 \times \Delta_1 + w_2 \times \Delta_2 + w_3 \times \Delta_3$$
$$where \quad w_1 > w_2 > w_3. \tag{6}$$

Since Δ_1 corresponds to situations where both objects definitely belong to the same cluster, the weight w_1 should have the highest value.

4 Simulated Annealing

Simulated Annealing [17], tries to overcome the local minima problems by incorporating probabilistic, rather than strictly deterministic approaches, in search of optimal solutions (see Fig. 2). We briefly overview the statistical mechanics aspects from which simulated annealing is conceived [18]. In thermodynamics, the probability of finding the system in a particular state with the energy E and temperature T is proportional to the Boltzmann probability:

$$e^{-\frac{E}{kT}} \quad where\ k\ is\ the\ Boltzmann\ constant. \tag{7}$$

Consider two state S_1 and S_2 with energy E_1 and E_2 , and the same temperature T . The ratio of probabilities of the two states is as following:

$$e^{\frac{[E_1 - E_2]}{kT}} \tag{8}$$

In our proposed method, the process of simulated annealing is as follows:

1. Randomly select solution vector x for the 0^{th} iteration. Set T to T_0.
2. Compute x_p a perturbed solution of x. x_p maybe obtained by randomly swapping two instances in x. Determine $\Delta E = E(x_p) - E(x)$.
3. Case 1. If $\Delta E < 0$, i.e., x_p is a better solution than x then select x_p as new x for the next step.

Case 2. If $\Delta E \geq 0$, select x_p with

$$e^{\frac{-\Delta E}{T}}$$

probability, keep current x with

$$1 - e^{\frac{-\Delta E}{T}}$$

Probability.

4. Repeat step 2 and 3 until $|\Delta E|$ is small enough.
5. Reduce T by, for example. $T_{new} = 0.9 \times T_{current}$. Repeat step 2 through 4. Terminate the entire process when T reduces zero or a small number.

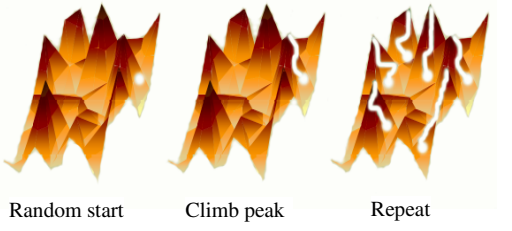


Fig. 2. Simulated Annealing searching process

In step 5 of the process, we reduce the temperature T, that is, we perform annealing. There is a trade-off between the reducing speed and finding the optimal solution. If the cooling speed is slow the global minimum may be guaranteed, but in this paper, to guarantee the global minimum a very slow reducing scheme is proposed as following:

$$T_t = \frac{T_0}{\log(1+t)} \quad \textit{where t is the } t^{th} \textit{ iterate.} \tag{9}$$

5 Optimized Rough SOM Using Simulated Annealing

We had successful experiments on our previous work Rough SOM [3], but to assign overlapped data to true cluster they belong to, the simulated annealing is employed. In this paper rectangular grid is used for the SOM. Before training process begins, the input data will be normalized. This will prevent one attribute from overpowering in clustering criterion. The normalization of the new pattern $X_i = \{x_{i1}, ..., x_{id}\}$ for $i = 1,2,...,N$ is as following:

$$X_{ij} = \frac{X_{ij} - \min(att_j)}{\max(att_j) - \min(att_j)}. \tag{10}$$

Once the training phase of the SOM neural network completed, the output grid of neurons which is now stable to network iteration, will be clustered by applying the rough set algorithm as described in the previous section. The similarity measure used for rough set clustering of neurons is Euclidean distance (the same used for training the SOM). In this proposed method (see Fig. 3) some neurons, those never mapped any data are excluded from being processed by rough set algorithm.

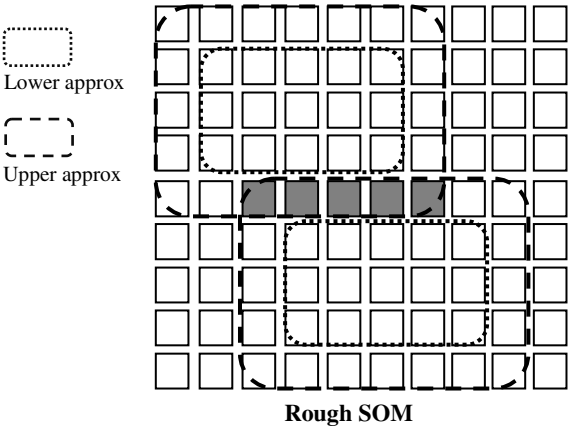


Fig. 3. Rough set clustering of the output grid of SOM

From the rough set algorithm it can be observed that if two neurons are defined as indiscernible (those neurons in the upper approximation of two or more clusters), there is a certain level of similarity they have with respect to the clusters they belong to and that similarity relation has to be symmetric. Thus, the similarity measure must be symmetric.

According to the rough set clustering of the SOM, overlapped neurons and respectively overlapped data (those data in the upper approximation) are detected. In the experiments, to calculate errors and uncertainty, the previous equations (5) and (6) will be applied to the results of the SOM (clustered and overlapped data). The n overlapped neurons and the distances to each m cluster centers they belong to (overlapped) could be represented as the following matrix:

$$M = \begin{array}{c} \overset{\longleftarrow \textit{Exsiting Clusters} \longrightarrow}{} \\ \begin{bmatrix} d_{11} & d_{12} & . & . & d_{1m} \\ d_{21} & d_{22} & . & . & d_{2m} \\ . & . & . & . & . \\ . & . & . & . & . \\ d_{n1} & . & . & . & d_{nm} \end{bmatrix} \end{array}$$

The distance between neuron i and the cluster centre j is determined as d_{ij} in matrix M. Simulated annealing is then applied to the set of distances to minimize the

(11) which optimize the clustering operation by assigning the nearest cluster centre to the overlapped neurons. The possible optimum-selected vector $x = (v_1, v_2, ...v_n)$ could be as $(d_{31}, d_{42}, ...d_{pq})$, which minimize the energy function as given as following:

$$F = \sum_{i=1}^{n} \sum_{j=1}^{m} d_{ij} \tag{11}$$

After the algorithm terminated, the clustering scheme would be as Fig. 4. The overlapped data are assigned to true clusters they belong to efficiently.

The aim of the proposed clustering approach is making the simulated annealing, which is applied to the rough SOM to be as precise as possible. Therefore, a precision measure needs to be used for evaluating the quality of the proposed approach. A possible precision measure can be defined as the following equation [15]:

$$certainty = \frac{\text{Number of objects in lower approx}}{\text{Total number of objects}} \tag{12}$$

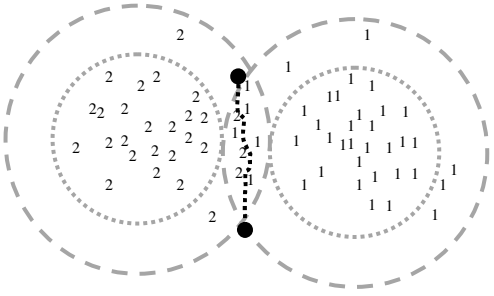


Fig. 4. The new border for clusters limit is highlighted, which shows the efficient assignment of the overlapped data between two clusters

6 Experiments Results

To demonstrate the effectiveness of the proposed clustering algorithm SA-RSOM (Rough set Incremental clustering of the SOM using Simulated Annealing), two phases of experiments has been done on two data sets, one artificial and one real-world data set.

The first phase of experiments presents the certainty that comes from the both data sets and in the second phase the errors has been generated. The results of SA-RSOM are compared to RI-SOM [4] and I-SOM (Incremental clustering of SOM) [19]. The input data are normalized such that the value of each datum in each dimension lies in $[0,1]$. For training, SOM 10×10 with 100 epochs on the input data is used. The general parameters for the SA algorithm have been configured as table 1.

The artificial data set has 569 data of 30 dimensions, which is trained once with I-SOM and once with RI-SOM then at last trained with SA-RSOM. The generated

Table 1. The general parameters of the simulated annealing algorithm

T_0	*0.65*
Number of Steps	*100*
Decrement Ratio	*0.9*
Boltzmann Probability (k)	*Random in (0,1)*
The Ratio of Probability according to equation (11)	$e^{\frac{[F_y - F_x]}{kT}}$

certainty (Fig. 5) is gained by the equation (12). From the table 2, it could be observed that the certainty-level in clustering prediction of SA-RSOM is more accurate compare to RI-SOM and I-SOM.

Table 2. The certainty-level generated by I-SOM, RI-SOM and SA-RSOM on the artificial data set, from epoch 100 to 500

Epoch	**100**	**200**	**300**	**400**	**500**
I-SOM	*56.29*	*67.78*	*79.91*	*91.29*	*92.51*
RI-SOM	*72.23*	*78.96*	*84.32*	*94.33*	*98.21*
SA-RSOM	*82.49*	*85.45*	*87.01*	*96.75*	***98.65***

The second data set is Iris data set from the UC Irvine Machine Learning Repository Database [20], has been widely used in pattern classification. It has 150 data points of four dimensions. The data are divided into three classes with 50 points each. The first class of Iris plant is linearly separable from the other two. The other two classes are overlapped to some extent. Fig. 6 shows the certainty generated from epoch 100 to 500. From the gained certainty, it is obvious that the SA-RSOM could efficiently detect the overlapped data that have been mapped by overlapped neurons (table 3).

In the second phase, the same initialization for the SOM has been used. Our proposed algorithms have generated the errors (table 4) that come from both data sets, according to the equations (5) and (6). The weighted sum equation (6) has been configured as following:

$$\sum_{i=1}^{3} w_i = 1 \quad Subject\ to: \quad w_i = \frac{1}{6} \times (4 - i). \tag{13}$$

Table 3. The certainty-level generated by I-SOM, RI-SOM and SA-RSOM on the Iris data set, from epoch 100 to 500

Epoch	**100**	**200**	**300**	**400**	**500**
I-SOM	*33.33*	*65.23*	*76.01*	*89.47*	*92.01*
RI-SOM	*67.07*	***73.02***	*81.98*	*91.23*	*97.33*
SA-RSOM	*69.35*	*72.54*	*82.26*	*95.27*	***98.05***

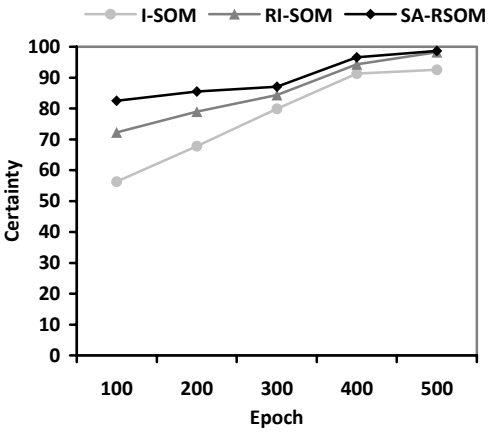


Fig. 5. The Comparative results for the artificial data set, from epoch 100 to 500

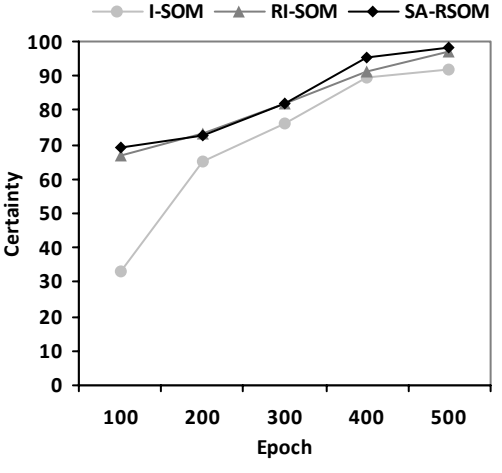


Fig. 6. The Comparative results for the Iris data set, from epoch 100 to 500

Table 4. The comparative generated errors

	Method	Δ_1	Δ_2	Δ_3	Δ_{total}
Artificial Data set	***SA-RSOM***	*0.6*	*0.88*	*0.04*	***1.4***
	I-SOM				*1.8*
Iris Data set	***SA-RSOM***	*1.05*	*0.85*	*0.043*	***1.94***
	I-SOM				*2.8*

7 Conclusions and Future Works

In this paper a two-level based clustering approach (SA-RSOM), has been proposed to predict clusters of high dimensional data and detect the uncertainty that comes from the overlapping data. The approach is based on the rough set theory that employs a soft clustering which can detects overlapped data from the data set and makes clustering as precise as possible, then SA is applied to find the true cluster for each overlapped data. The results of the both phases indicate that SA-RSOM is more accurate and generates fewer errors as compare to crisp clustering (I-SOM).

The proposed algorithm detects accurate overlapping clusters in clustering operations. As the future work, the overlapped data also could be assigned correctly to true clusters they belong to, by assigning *fuzzy membership value* to the indiscernible set of data. In addition, a weight can be assigned to the data's dimension to improve the overall accuracy.

Acknowledgements. The Research Management Centre, University Technology Malaysia (UTM) and the Malaysian Ministry of Science, Technology and Innovation (MOSTI) supported this research under vote number 79224.

References

1. Kohonen, T.: Self-Organized formation of topologically correct feature maps. Biol. Cybern. 43, 59–69 (1982)
2. Kohonen, T.: Self-Organizing Maps. Springer, Berlin (1997)
3. Mohebi, E., Sap, M.N.M.: Hybrid Self Organizing Map for Overlapping Custers. In: Springer-Verlag Proceedings of the CCIS, Hainan Island, China (accepted) (2008)
4. Mohebi, E., Sap, M.N.M.: Rough set Based Clustering of the Self Organizing Map. In: IEEE Computer Scociety Proceeding of the 1st Aseian Conference on Intelligent Information and Database Systems, Dong Hoi, Vietnam (accepted) (2008)
5. Gray, R.M.: Vector quantization. IEEE Acoust. Speech, Signal Process. Mag. 1(2), 4–29 (1984)
6. Pal, N.R., Bezdek, J.C., Tsao, E.C.K.: Generalized clustering networks and Kohonen's self-organizing scheme. IEEE Trans. Neural Networks (4), 549–557 (1993)
7. Han, J., Kamber, M.: Data mining: concepts and techniques. Morgan Kaufmann, San Francisco (2000)
8. Asharaf, S., Narasimha Murty, M., Shevade, S.K.: Rough set based incremental clustering of interval data. Pattern Recognition Letters 27, 515–519 (2006)
9. Yan, Yaoguang: Research and application of SOM neural network which based on kernel function. In: Proceeding of ICNN&B 2005, vol. (1), pp. 509–511 (2005)
10. Sap, M.N.M., Mohebi, E.: Outlier Detection Methodologies: A Review. Journal of Information Technology, UTM 20(1), 87–105 (2008)
11. Jain, A.K., Murty, M.N., Flynn, P.J.: Data Clustering: A Review. ACM Computing Surveys 31(3), 264–323 (1999)
12. Stahl, H.: Cluster analysis of large data sets. In: Gaul, W., Schader, M. (eds.) Classification as a Tool of Research, pp. 423–430. Elsevier North-Holland, Inc, New York (1986)
13. Can, F.: Incremental Clustering for dynamic information peocessing. ACM Trans. Inf. 11(2), 143–164 (1993)

14. Sharma, S.C., Werner, A.: Improved method of grouping provincewide permanent traffic counters. Transaction Research Report 815, Washington D.C., pp. 13–18 (1981)
15. Pawlak, Z.: Rough sets. Internat. J. Computer Inf. Sci. (11), 341–356 (1982)
16. Lingras, P.J., West, C.: Interval set clustering of web users with rough K-means. J. Intelligent Inf. Syst. 23(1), 5–16 (2004)
17. Larhoven, P.J.M., Aarts, E.H.L.: Simulated Annealing: Theory and Applications. Springer, Berlin (1987)
18. Munakata, T.: Fundamentals of the New Artificila Intelligence. Springer, Heidelberg (2008)
19. Sap, M.N.M., Mohebi, E.: A Novel Clustering of the SOM using Rough set. In: IEEE Proceeding of the 6th Student Conference on Research and Development, Johor, Malaysia (accepted) (2008)
20. UC Irvine Machine Learning Repository Database (1987), `http://archive.ics.uci.edu` (accessed April 12, 2008)

Interactive Quality Analysis in the Automotive Industry: Concept and Design of an Interactive, Web-Based Data Mining Application

Steffen Fritzsche[1], Markus Mueller[2], and Carsten Lanquillon[3]

[1] Ulm University, Institute of Applied Information Processing, Ulm, Germany
steffen.fritzsche@uni-ulm.de
[2] University of Bamberg, Laboratory for Semantic Information Technology, Bamberg, Germany
markus.mueller@uni-bamberg.de
[3] Heilbronn University, Institute of Electronic Business, Heilbronn, Germany
lanquillon@hs-heilbronn.de

Abstract. In this paper we present an interactive, web-based data mining application that supports quality analysis in the automotive industry. Our tool is designed to help automotive engineers in their task of identifying the root cause of quality issues. Knowing what exactly caused a problem and identifying vehicles that are most likely to be affected by the issue, helps in planning and implementing effective service actions. We show how data mining can be applied in the given application domain, point out the key role of interactivity and propose an appropriate software architecture.

Keywords: Data Mining, Web Interfaces and Usability, Interactivity, Automotive Quality Analysis.

1 Introduction

Modern vehicles are highly complex, mechatronic systems. To ensure top quality, automotive manufacturers improve their processes to avoid design issues from the beginning, to guarantee top quality during the assembly process, and finally, to get problems detected and fixed as fast as possible. A quality feedback of discovered issues from After Sales to engineers in Design and Manufacturing closes the loop and is key to a continuous improvement of quality. In this paper we review the concept of interactive data mining and propose an architecture for an interactive data mining tool that is designed to help automotive engineers in their task of identifying the root cause of quality issues. Knowing what exactly caused a problem and identifying subsets of vehicles that are most likely to be affected by the issue, helps in planning and implementing effective service actions.

Consider the following example: For some reason a light in the instrument cluster indicates a problem with the engine. Customers are concerned, but the dealer cannot identify any serious problem and simply clears the trouble code. In several cases, dealers might erroneously replace sensors or other engine parts.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 402–414, 2009.
© Springer-Verlag Berlin Heidelberg 2009

Applying data mining, a quality engineer can search for a statistical regularity in the data. He might find out that the failure code is more likely to be set on vehicles that are equipped with a specific hard and software version of the engine control module than on other vehicles. Based on this information, the module can be checked, a software error can be fixed, and during the next regular maintenance schedules, the software of all affected vehicles can be updated.

Designing and implementing a data mining application with innovative analysis functionality that supports this type of causal investigations is quite challenging: First, most often the true cause of quality issues is not among the available influence variables and significant influences can only be indicators for the true, but *hidden* variable. Therefore, our goal can only be to come as close to the root cause as possible. Second, for many quality issues, the class distribution is very imbalanced as the number of vehicles with a specific quality problem, the so called *non-conforming* vehicles, is quite small. Third, data quality is a major issue. As an example, class labels may be imprecise and vehicles that are declared non-conforming might actually be conforming. What is more, previous service actions or production clean points can result in heterogeneous problem settings. Yet another important challenge is that the users have to be convinced that data mining provides valuable insights and really is a solution to their problems.

In our experience, the most effective way to deal with all these issues is to design the data mining application to support an *iterative*, *interactive*, and *intuitive* investigation. A user gains insight into the business problem at hand during the interactive process of creating a data mining model rather than from the bare results. Besides the end user will have more confidence in the patterns found if he understands the process of pattern generation and if he is part of the process. This is essential to increase user acceptance and to make the system valuable.

The concept of an interactive, web-based data mining application is innovative with respect to several aspects. Traditional data mining applications are either desktop tools that are used by data analysts and expert users, or run as automated batch jobs on mainframes with preset parameters. Some more recent commercial solutions do provide basic data mining functionality within web clients. However, these approaches do not yet support truly interactive modeling as is required for the task of quality analysis and they mostly lack the ability to incorporate user-defined modules such as alternative scoring or evaluation functions. Finally reaching a scalable architecture for an interactive web application was the result of an evolutionary process. Hence, apart from the final architecture, we want to present lessons-learned and pitfalls when migrating from a research desktop tool to a scalable, interactive web application for data mining.

This paper is organized as follows. In the next section we describe the data mining task at hand in more detail. This section also reveals the specific requirements to the system from a design perspective. The evolution of our system and a rough description of our proposed architecture can be found in Section 3. The primary pile of the architecture, an innovative presentation framework for interactive web applications, is described in Section 4. The second pile, a scalable mining backend is explained in Section 5.

2 Interactivity Closes the Gap

According to a standard definition, "Knowledge Discovery in Databases (KDD) is the non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data" [5]. In our context a *pattern* describes the characteristics of a subset of *non-conforming* vehicles that share a quality issue. Vehicle characteristics include hundreds of variables like model, engine type, production date, or sales codes that are pulled from a quality data warehouse which is based on various operational data sources such as production data, warranty claims and diagnostics information from the dealership.

A *valid* pattern is correct and statistically significant, but whether it is actually *new* and *useful* to a business expert, mainly depends on his background knowledge. Several research attempts have been made to incorporate this knowledge into the process of finding and assessing patterns, e.g. in [8]. However, in practice one often faces the problem that the process of knowledge modeling is very elaborate and that many business experts are neither willing nor able to make their knowledge explicit (*knowledge acquisition bottleneck*).

In our experience interactivity can close this enormous gap between what data mining methods can deliver and what is actually needed for solving a problem. By "drilling through the patterns" the expert can get a better understanding of what is actually going wrong although the true cause might be hidden to the data mining system. Note that this is contrary to many other common data mining tasks. Consider, for example, the task of predicting some future behavior of certain objects, such as customers and their affinity to buy a product as you will find it in many marketing applications. Here, it is most important to create a very accurate model whereas comprehensibility of this model is not more than a nice side effect in order to better understand customer behavior.

2.1 Interactive Decision Trees

Apart from other data mining methods, interactive decision trees [11], [1] are applied as they provide a very intuitive and efficient way to split cases in a database into subsets of similar cases by minimizing or maximizing a given split function. Common split algorithms minimize the *average class entropy* [4], maximize the *information gain* [11], or optimize the trade-off between *precision* and *recall*.

In our application domain, a single case or *instance* is a vehicle with specific vehicle characteristics. Each instance is assigned a class label to separate conforming from non-conforming vehicles. A node in a decision tree refers to a subset of vehicles with specific properties and shows the number of vehicles for each class label. In each step of splitting a node into sub nodes, an attribute is selected that is used to separate the non-conforming vehicles from the conforming ones.

Assume the following illustrating example: 115 in 6000 vehicles are non-conforming vehicles that are brought to a dealership because a lamp indicates a diagnostic trouble code (DTC). The data shows a significant deviation in the fault rate for Sedan vehicles with seat heating (Table 1). A corresponding decision tree is illustrated in Figure 1. The node color indicates the strength of the deviation of the fault rate of a sub group from the overall fault rate.

Table 1. Example dataset showing the issue counts and the number of all vehicles for various configurations. 115 in 6000 vehicles are non-conforming vehicles with a specific DTC set.

Seat Heating	Body Style	issue	sum
Yes	Sedan	70	1000
No	Sedan	15	2000
Yes	Coupe	20	2000
No	Coupe	10	1000
		115	6000

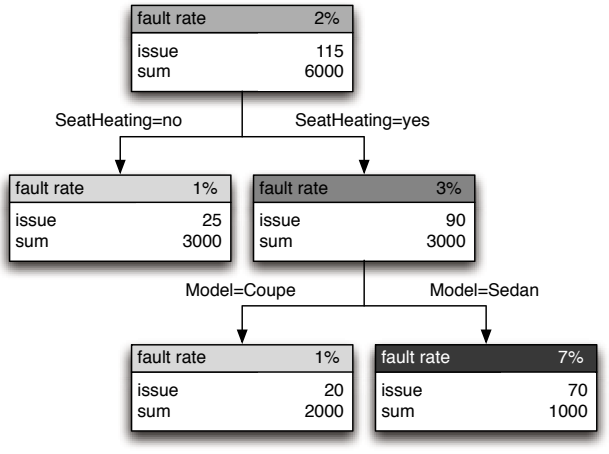


Fig. 1. Decision tree for the example dataset: The fault rate is extraordinarily high for sedans with seat heating

Although finding the best decision tree fitting the training data is a standard data mining task, in practice the key is interactivity: The decision tree algorithm has to recommend a list of the most significant split attributes, but the engineer should have the freedom to actually pick a split variable manually even if it is ranked low. Such an interactive analysis allows the user to develop, test and reject hypotheses about the root cause of a failure. In this case, even finding *nothing* significant can be a very helpful result as it avoids unnecessary investigations. What is more, as the solution space grows exponentially and as the true cause might not even be among the available variables, it is very likely that an automatically generated tree is meaningless. If a user interacts with a system and thus guides an heuristic search algorithm, he obtains results that might not necessarily maximize a statistically motivated scoring function, but that will maximize his personal importance score.

2.2 Other Interactive Mining Tasks

Decision trees are very well suited for interative mining as they do not require too much computational effort because split quality of all available attributes

has to be computed only for the one node that the user currently considers. Yet, there are some drawbacks as they allow to follow only one interesting path at once and commonly perform only univariate splits.

To mitigate these issues other mining algorithms such as directed rules obtained from frequent item sets or rule cubes [2] have been developed. In contrast to the decision tree approach, these methods require intensive and long running computations to generate a result set based on which interactive pattern exploration may start.

For an interactive tool which also contains some long running processes, it is crucial that the user gets informed about the current status of these processes, including an estimate of the time remaining to complete the task. In addition, the user should be able to continue other threads of his analysis and review previously done calculations and configurations while complex and long running processes are executed.

3 Architecture

The specific requirements of the application domain described in the last sections and in particular the highly interactive approach must be considered when designing a proper software architecture. Our proposed architecture for a scalable, interactive web application for data mining was the result of an evolutionary process. Hence, apart from the final architecture, we want to present design issues and lessons-learned when migrating from a research desktop tool to an interactive web application.

3.1 Evolution of the Architecture

The architecture of our initial desktop application with local database is depicted in Figure 2. There exists a central data warehouse that contains some quality data, but many other data sources have to be integrated for specific mining tasks. In a manual ETL process, data is extracted, transformed and loaded into a local database installed on the desktop client. By joining tables, selecting rows and analysis features, and by transforming rows into columns, the client application allows the derivation of a user-defined dataset that is loaded into main memory. This preprocessing allows a fast execution of various mining tasks. For example, the recommendation of split attributes or the creation of the most likely decision tree can be done very efficiently. Besides, the dataset can be narrowed down and extracted sub-datasets can be made available to other mining tasks. Furthermore, multi-threading allows the processing of long-running mining tasks, like the task of extracting association rules, in the background, while the user can continue in an interactive analysis. As main memory and high-performance desktop PCs are no longer very expensive, the system can be scaled up very easily. Moreover, using a powerful API like Java Swing, allows the design of an intuitive and user-friendly graphical user-interface that supports a fast, interactive exploration of the data and the corresponding hypothesis space.

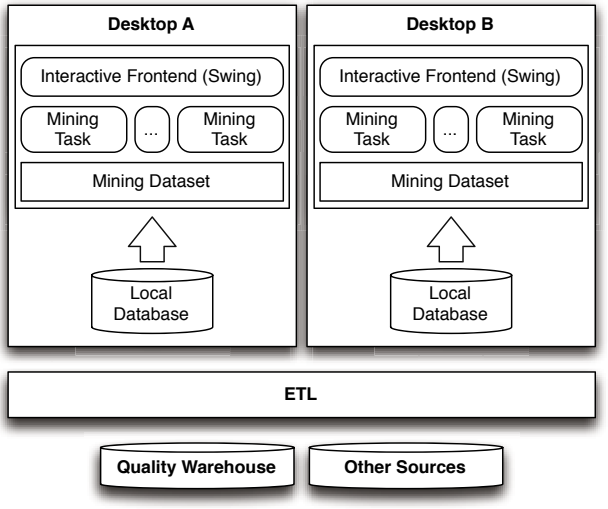


Fig. 2. Architecture scheme of the original research desktop client

The most important drawback of such an architecture is the decentralized data storage. A non-standardized ETL process is not only time-consuming, but also causes data quality issues. What is more, there is no longer a *single point of truth* as many data islands exist. An analysis done on one desktop might produce different results on another one. Bookkeeping of the respective datasets is very difficult. Besides, quality data is highly confidential and distributing this data arbitrarily causes privacy and data protection issues. As a consequence, in modern IT infrastructures, data is kept in one central data warehouse. In such an environment, a user creates an analysis dataset which is transferred over the network and available for the time of an analysis session. This leads to a consistent view on the data and protecting the data is easier. To reduce network load, a local caching and updating of datasets is necessary.

Another major goal of IT departments is to cut maintenance costs by reducing the administrative effort for hundreds of thousands of desktop computers. Each additional desktop software requires keeping track of current software versions, required security updates, or the roll-out of new features. In contrast, modern web applications are easy to maintain and clients only require standard software, in particular a web browser with an efficient scripting engine. When new features are developed, they can be deployed easily by upgrading the software on a few servers. This is a very important feature for a research system like ours that continuously undergoes changes and improvements. In particular, extensions like new mining methods or better algorithms can easily be rolled out. The system is widely available to power users and to infrequent users without expensive installation.

3.2 Interactive, Scalable Web Application

The rough design of a data mining web application is straightforward: Data is kept in a central data warehouse and is accessed by a web application

running in an application server. Several user-defined datasets can be derived by joining data, selecting rows and features, and by transforming the data. This solution seems to solve our problems: Data is only transferred from a database server to an application server which are both connected by a fast and direct connection. Authentication, authorization and preventing illegal access, e.g. by SQL-injection, can be assured by the web application. Finally, new features can simply rolled out by upgrading the software on a few servers.

However, realizing an appropriate architecture is challenging. On the one hand, interactivity is key to the overall application workflow and therefore it is necessary to create a presentation layer that supports this. Besides, the mining backend system must be able to deal with huge datasets and long-running, computationally demanding data mining tasks. The main components of our proposed architecture are depicted in Figure 3. An important step towards better scalability is to split up the web application and to deploy it on several servers. In the simplest case, there is one web application server that coordinates the interaction between the client and the server and forwards requests to mining tasks that run in a single mining backend server. The frontend application and the backend application communicate by *Message Queueing*. This has two advantages: First, the frontend server is not slowed down, when complex mining tasks are run in the backend, and second, this scalable approach allows to increase the number of backend servers that do the data handling and run the

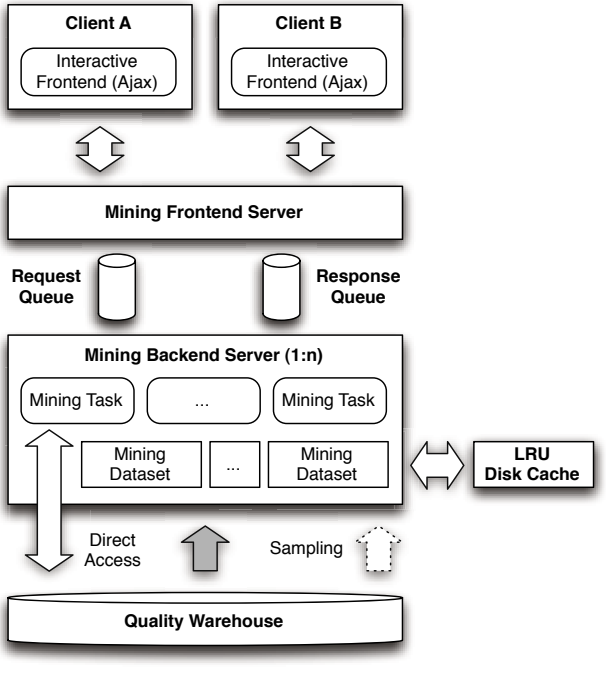


Fig. 3. Architecture scheme of an interactive, scalable web application

mining tasks. Queueing requests has the following advantages: First, the number of worker threads that empty the queue and process requests can be configured easily. Thus, it is easily possible to adjust the maximum number of concurrent mining processes. Besides, a scheduler can be applied to make sure that tasks are prioritized properly. An even simpler solution that we apply, is to have two request queues, one for long running mining tasks, and another one for relatively fast database queries. The results are written back to the response queue. The presentation archictecture and specifics of the mining backend are explained in the following two sections.

4 Interactive Web-Based Presentation Layer

Interactivity is key to the overall application. Several additional functional and non-functional requirements are driving the presentation architecture decisions. First of all, it is necessary to support rich user interface (UI) components like the ones needed to represent the decision trees. Moreover, these components are not designed to be view-only. Rather, they are the key navigation and interaction elements of the application. Many UI events originating fairly complex components have to be taken into account during the design process of the resulting web interface. Beside all these interactivity issues, the presentation layer has to support the potentially long-running data loading and mining tasks. It is important that the user is always informed about the current processing state and running processes should not block other parts of the presentation layer. The user should be able to continue the analysis or review previously done calculations and configurations while complex and long-running processes are executed on the server. No action or event should ever break the users' workflow. To implement such an highly interactive web application, we apply asynchronous Javascript and XML (Ajax) together with standard compliant web-technologies like XHTML and CSS. Nevertheless, it should be possible to replace this specific view implementation with another one and reuse the server-side view logic, e.g. to provide an additional swing GUI for power-users.

In contrast to many other applications, all user interactions and the resulting view make up the final result of the users' analysis sessions. Due to the fact that such an analysis session can easily last several hours, the presentation layer must always preserve the generated view state. No matter what happens to the presenting client, it should be possible to reconstruct this view at any time. Besides, the presentation framework should separate the underlying application flows from the actual client-view technology to make reuse as easy as possible if it becomes necessary to change the view technology.

Most presentation frameworks (e.g., Java Server Faces, Spring MVC) are too tightly coupled with a certain client presentation technology or there is no support for the highly interactive components. The traditional Model View Controller (MVC) based architecture as described in [9] is not the best choice for our application. In particular, it is hard to decouple the client view technology because of the dependencies between model and view in traditional MVC. Some

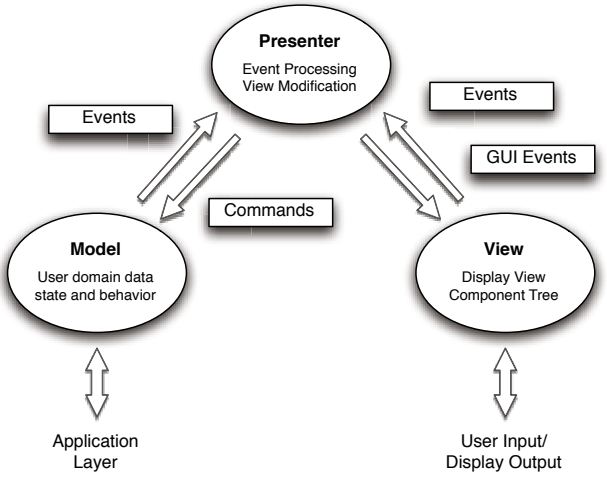


Fig. 4. Implemented MVP solution based on the Passive View Pattern

modifications of the traditional MVC pattern are outlined in [3]: The controller gets more intelligent and is allowed to perform view modifications. However, the less intelligent the client view, the easier it is to replace a special client view implementation and to keep an up-to-date view state on the server. To accomplish this, it is necessary to break any dependencies between model and view. Hence, the Model View Presenter (MVP) pattern as outlined in [10] is preferable. From the two MVP architecture styles described in [6], the *Passive View* pattern fits our requirements best.

Figure 4 shows a simplified model of our passive view implementation. The original pattern had to be tweaked in several aspects to match our specific requirements. There is no longer a dependency between the presentation model and the view. Moreover, the view implementation is quite dumb. Native UI events are caught and translated into custom framework events. This step is necessary to decouple the presentation framework from the client technology used. The presenter receives the translated events and executes the associated predefined commands to modify the backing model. Each model change results in an event which is passed back to the presenter. Based on this model change event the presenter gathers the necessary view changes from the backing configuration to adjust the view state.

The presenter is the core of our presentation framework. Internally, a *Configuration Engine* reads and verifies the framework's XML based configuration files, especially component and event files, during startup. Furthermore, an *Event Processor* receives the view and model events and translates them into framework commands. According to the configuration, the *Command Executer* is used to modify the model by calling the backend API. Each of these calls may result in new events. This chaining of events allows the construction of complex workflows and view modifications.

When a client event reaches the server-side presenter, it is placed in the user's event queue. At this point, the asynchronous client events are synchronized to avoid an inconsistent view state caused by bad timing or network latency. The already introduced *Event Processor* picks up the queued events and executes the commands found in the event configuration. After command execution, the presenter performs the specified view modifications. Currently the framework supports six types of modification actions: adding, removing, hiding, showing and moving a certain component as well as clearing the entire view tree. Moreover, all modifications on the underlying model are then pushed into the new or existing view components.

The server-side view representation is a hierarchical tree and is generated by nesting predefined components. The *Configuration Engine* defines which components are available and ready for use within the current application. Any client specific view has to provide a suitable implementation for each of the defined components. To construct a view, a view specific, configured version of the components is placed within the tree. These *Configured Components* contain all information needed by the presenter like binding information for data that should be pushed into the component, implementing client view classes and whether this component is providing extension points to host further components. Moreover, it is also possible to configure the removal of a *Configured Component* on the occurrence of certain events.

Due to the fact that there should be no dependencies between a component and its data stored in the model each component configuration names the possible "data push identifier". These IDs are specific to a view component, but not to the components' client implementations. The presenter monitors the model and pushes any changed data for currently visible components directly into to the view using the predefined binding information and the components data push IDs. When and how these data changes reach the actual client view depends on the used client technology. These changes might be queued until the client state is synchronized. Each user has its own model instance stored within the user's session object. Due to the fact that this application may have to display large amounts of analysis data and analysis models, it is necessary to make sure all references to data stored within the user session are released as soon as possible. To accomplish this, only data for visible components is hold within the session. References to unused data are released by comparing the predefined binding information with the current server-side view tree.

On the client side, some of the MVP-framework parts are rebuilt. The use of the same concepts on both sides, the generic framework and the specific client implementation, make the process of view generation and synchronization easier. The current Javascript-based client implementation consists of four main parts. A *View Tree* which is a simplified version of the framework's hierarchical view tree, only containing the information about the component hierarchy and the Javascript client specific implementation classes. The second part is the client-side *Data Distributor*. Theoretically, the server-side presenter should push all data changes directly into the components. Due to the fact that the HTTP

protocol is stateless, however, there is no way for our server to get back to the client. This is the reason why the *Data Distributor* performs a server poll every few seconds to request changed data from the server. A client-side *Event Dispatcher* is the third pile used to catch all native HTML events originating HTML elements belonging to a certain logical framework component. It translates these native events to framework events and registers appropriate callback methods to handle the asynchronous event processing. Each part represents a small portion of the server-side presenter whose API is made available with a traditional proxy-based approach.

5 Scalable Mining Backend

The main challenge for the mining backend is the size of the analysis datasets that are kept in main memory. Automatic feature selection and sampling is applied to reduce the size of the datasets without losing too much information. Moreover, an efficient memory management is implemented.

Feature selection aims at deriving a subset of attributes that are highly relevant for an analysis. A general introduction on the topic can be found in [7]. Consider a simple feature selection for decision trees: Any attribute with only one value cannot be used as split attribute and thus can omitted. A more sophisticated feature selection calculates the dependency between the influencing variables and the binary target variable and selects these attributes that seem to be the most relevant in a subsequent, detailed analysis.

Sampling is another technique that is generally applied to scale up data mining algorithms. Mining tasks produce approximate results when run on randomly selected data. The accuracy of the results mainly depends on the sample size and can be estimated by calculating upper and lower confidence bounds for the unknown parameter. As the number of non-conforming vehicles is very small in our domain, simple Bernoulli sampling would result in a dataset that might not even contain a single non-conforming vehicle. Hence, one has to apply an extreme form of stratified sampling: One chooses all non-conforming vehicles and samples only from the subset of vehicles without the quality issue that is currently investigated. To derive a final decision tree with exact numbers, the tree that was created on sample data, can be simply rebuilt by querying the database. Note that this is a far less complex task than the task of initially growing this tree. More sophisticated approaches like *sequential sampling* allow for precise bounds on the confidence of solutions, while keeping the number of database queries small [12].

Memory management is improved by the introduction of a least recently used (LRU) in-memory / disk cache. The cache allows to configure the maximum amount of main memory that should be used to store the variables of the analysis datasets. While most open source cache implementations for Java (e.g., EHCache, JCache) start swapping the least recently used objects to disk when a *maximum number of elements* in the cache is reached, one rather wants to limit the *amount of memory used*. As a dataset mainly consists of arrays of primitive

datatypes, the amount of memory that is occupied, depends on the number of records in the dataset and the datatype of the attribute. Now, when the cache starts swapping, a small array that is moved to disk could be replaced by a large array and the JVM would run out of memory. The solution is to split up the arrays into smaller chunks of equal size. The respective size depends on the datatype and is larger for smaller datatypes. The chunk size should be chosen to optimize the trade-off between reduced performance due to an increased number of read and write operations, and an increased amount of unused main memory if chunks are not filled up for small datasets.

At first glance the requirements of our application seem to be similar to the interactive OLAP approach and the question arises why data is kept in main memory at all. For example, all contingency table counts needed to create a decision tree could be simply obtained by SQL queries. The main difference is that our users work on the same dataset that can contain hundreds of attributes for a long time and explore it from various perspectives. In this case a flat, denormalized dataset that is kept in main memory has several advantages: Joining of data, flattening of attributes and imposing row and feature restrictions on the data has to be done only once. Unless one creates a temporary database table, this would have to be done again and again when using temporary views or auto-created SQL queries. In addition, counts can be obtained faster than by querying a huge database. One might argue that OLAP tools apply the idea of caching by precounting multidimensional aggregates. These tools exploit the fact that most reports are standard reports. Hence, an architect can predefine aggregates and the tool can gather statistics and can suggest further aggregates that lead to an increasing performance. However, an analysis dataset can contain hundreds of attributes and one cannot precalculate three, four or sometimes even five dimensional aggregates for all possible combinations of attributes. Which attributes are most important, depends on the specific analysis scenario. Simply restricting the number of allowed rows and features to reduce the size of the datasets is not satisfying either. The user would have to narrow down the dataset manually, but if a user already knew what instances and what features are primarily relevant, he would not need a data mining system.

6 Conclusion and Future Work

In this paper we introduced an application that supports quality analysis in the automotive industry. We pointed out that interactivity plays a key role to close the gap between data mining algorithms and practical requirements. When designing a proper software architecture this interactivity and long running, computationally expensive, and memory intensive data mining processes have to be considered. We describe the evolution of our software architecture, discuss pros and cons of various milestones and finally present the architecture of an interactive, scalable web application. This architecture consists of an interactive, web-based presentation layer and a scalable mining backend.

The Model View Presenter framework described in this paper is designed as a response to the specific presentation requirements. It contains a robust, reusable

server-side presentation backend that is capable of serving view data for many different client view technologies. One possible client view implementation to realize a light-weight web client, based on Ajax and standard-compliant web technologies, is outlined in this paper. The mining backend proves to be robust and efficient due to sampling, feature selection and caching.

Finding a way to handle large datasets when sampling is not possible is a future research task. From a practitioner's perspective, a promising way is to find an optimum between caching of data and querying the database. Consider the creation of a decision tree: After the first split, the size of a dataset is often reduced tremendously. Exploiting this fact, one could calculate splits for the first level by querying the database, e.g., by running a stored procedure, and calculate splits for the next levels on a relatively small dataset that is kept in main memory. Another task is to integrate other interactive data analysis methods into the framework.

References

1. Blumenstock, A., Hipp, J., Kempe, S., Lanquillon, C., Wirth, R.: Interactivity closes the gap. In: Proceedings of the KDD 2006 Workshop on Data Mining for Business Applications, Philadelphia (2006)
2. Blumenstock, A., Schweiggert, F., Mueller, M.: Rule cubes for causal investigation. In: Proceedings of the Seventh IEEE International Conference on Data Mining, Philadelphia (2007)
3. Buschmann, F., Meunier, R., Rohnert, H., Sommerlad, P., Stal, M.: Pattern-Oriented Software Architecture: A System of Patterns, Model-View-Controller, pp. 125–143. John Wiley & Sons, Chichester (1996)
4. Elomaa, T., Rousu, J.: General and efficient multisplitting of numerical attributes. Machine Learning 36, 201–244 (1999)
5. Fayyad, U., Piatetsky-Shapiro, G., Smyth, P.: From data mining to knowledge discovery. In: Fayyad, U., Piatetsky-Shapiro, G., Smyth, P., Uthurusamy, R. (eds.) Advances in Knowledge Discovery and Data Mining. AAAI/MIT Press, Cambridge (1996)
6. Fowler, M.: Passive View (2006), http://www.martinfowler.com/eaaDev/PassiveScreen.html
7. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. Journal of Machine Learning Research 3, 1157–1182 (2003)
8. Jaroszewicz, S., Simovici, D.A.: Interestingness of frequent itemsets using bayesian networks as background knowledge. In: KDD 2004: Proceedings of the 2004 ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 178–186. ACM Press, New York (2004)
9. Krasner, G.E., Pope, S.T.: A cookbook for using the model-view controller user interface paradigm in smalltalk-80. J. Object Oriented Program. 1(3), 26–49 (1988)
10. Potel, M.: MVP: Model-view-presenter, the taligent programming model for c++ and java (1996), http://www.wildcrest.com/Potel/Portfolio/mvp.pdf
11. Quinlan, J.R.: C4.5: Programs for Machine Learning. Morgan Kaufmann, San Mateo (1993)
12. Scheffer, T., Wrobel, S.: Finding the most interesting patterns in a database quickly by using sequential sampling. Journal of Machine Learning Research 3, 833–862 (2002)

NARFO Algorithm: Mining Non-redundant and Generalized Association Rules Based on Fuzzy Ontologies

Rafael Garcia Miani, Cristiane A. Yaguinuma, Marilde T.P. Santos, and Mauro Biajiz

Department of Computer Science, Federal University of São Carlos (UFSCar)
P.O. Box 676, 13565-905 São Carlos, Brazil
{rafael_miani,cristiane_yaguinuma,marilde,mauro}@dc.ufscar.br

Abstract. Traditional approaches for mining generalized association rules are based only on database contents, and focus on exact matches among items. However, in many applications, the use of some background knowledge, as ontologies, can enhance the discovery process and generate semantically richer rules. In this way, this paper proposes the NARFO algorithm, a new algorithm for mining non-redundant and generalized association rules based on fuzzy ontologies. Fuzzy ontology is used as background knowledge, to support the discovery process and the generation of rules. One contribution of this work is the generalization of non-frequent itemsets that helps to extract important and meaningful knowledge. NARFO algorithm also contributes at post-processing stage with its generalization and redundancy treatment. Our experiments showed that the number of rules had been reduced considerably, without redundancy, obtaining 63.63% average reduction in comparison with XSSDM algorithm.

Keywords: Data Mining, Generalized Association Rules, Redundant Rules, Fuzzy Ontology.

1 Introduction

Data mining is a key step of knowledge discovery in large databases [1]. One important topic in data mining research is concerned with the discovery of interesting association rules [2]. Many approaches on mining association rules are motivated by finding new ways of dealing with different attribute types or increasing computational performance. In the mean time, a crescent number of approaches have been developed regarding semantics of mined data, aiming at improving the quality of obtained knowledge. In this sense, ontologies have been widely employed to represent semantic information defined by knowledge experts, and can also be applied to enhance association rule mining. Some researches [3] [4] have extended the process of mining association rules in order to obtain rules that represent relation between basic data items, as well as between items at any level of the related taxonomy (is-a hierarchies) or ontology, resulting in the so called generalized association rules.

The use of traditional ontologies based on propositional logic (crisp ontologies), which considers exact reasoning discriminated on “false” or “true”, is a common

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 415–426, 2009.
© Springer-Verlag Berlin Heidelberg 2009

point in approaches to generalize association rules. Such restriction becomes inappropriate to represent some concepts and relationships of the real world. For example, it is difficult to represent in crisp ontologies concepts such as "young", "old", "high" or "low" as well as fuzzy relationships like the similarity relation [5], which has a degree representing the strength of how concepts are similar to each other. Hence, the combination of ontologies and fuzzy logics, based on the theory of fuzzy sets [6], is suitable to express the uncertainty inherent in specific domains.

Therefore, some researches have used fuzzy ontologies in order to extract semantically richer association rules [7] [8]. However, in general, by adopting either crisp or fuzzy ontologies in generalized association rule mining, the process of extracting association rules usually brings to users a great amount of rules that represent the same information. Thus, it is interesting to avoid mining unnecessary rules that express redundant knowledge, while focusing on the semantic richness provided by fuzzy ontologies.

Considering this context, this paper proposes the NARFO (Non-redundant Association Rule based on Fuzzy Ontologies) algorithm to mine non-redundant and generalized association rules based on fuzzy ontologies. The main contribution of our research is the generalization of non-frequent itemsets, in addition to the generalization of the extracted rules and the redundancy treatment, all considering fuzzy itemsets.

The remainder of this paper is organized as follows. In section 2, we present related work. Section 3 explains the NARFO algorithm and shows its main characteristics. Performed experiments are showed in section 4. Finally, section 5 brings some conclusions and future work.

2 Related Work

Many researches have been employing taxonomies and ontologies as background knowledge in mining association rules in order to enhance the knowledge discovery process. [4] considers domain knowledge to generalize low level rules discovered by traditional rule mining algorithms, in order to get fewer and clearer high level rules. ExCIS algorithm [9] applies domain knowledge in pre and post-processing steps. The preprocessing step uses ontology to guide the construction of specific datasets for particular mining tasks. In the post-processing step, mined rules are interpreted and filtered, as terms are generalized based on the ontology.

“However, in many real-world applications, the taxonomic structure may not be crisp but fuzzy.” [1]. For this purpose, [1] developed an algorithm to mine generalized association rules with fuzzy taxonomic structures. In addition to the minimum support (*minsup*) and minimum confidence (*minconf*) measures, the algorithm considers the *R-interest* measure, which is used to eliminate redundant and inconsistent rules. Fuzzy association rule mining, developed by [2], is driven by domain knowledge in order to make the rules more visual, more interesting and understandable. Database attributes are mapped as linguistic variables, which are divided into linguistic terms. For example, attributes like age, education and skill (linguistic variables) concern to the high-level concept person (linguistic term) of the ontology. XSSDM algorithm [8] proposes another approach that uses fuzzy ontologies to represent the semantic similarity relations among mined data. This algorithm considers a new measure, called minimum similarity (*minsim*). If two items have the similarity degree greater

than or equal the *minsim*, fuzzy associations are made and can be expressed in the association rules extracted by the algorithm. For example, if $item_1$ and $item_2$ have the degree of similarity greater than or equal *minsim* in the fuzzy ontology, a fuzzy association is made and a fuzzy itemset is created (represented as $item_1$~$item_2$).

Although generalized association rule mining approaches based on fuzzy ontology express semantically richer information, they may result in a great amount of redundant rules. Thus, redundancy treatment has been an interesting research topic. In [10] a multiple level association rule was proposed to reduce the number of generalized association rules. This consists in defining different *minsup* values for each level of a given taxonomy, in which higher levels have bigger *minsup* values. Some approaches focus on reducing the amount of generalized and redundant association rules during the pattern extraction process. The cSET algorithm [11] considers the concept of closed itemset [12]. The MFGI_class algorithm [13] is based on maximal frequent itemset theory [14]. There are other researches that treat the problem after the processing stage. [1] proposes the generalization process based on the *R-interest* measure, which prunes redundant rules, only considering the rules whose degree of support and confidence are *R* times the expected degree of support or confidence. So, the rules are generalized if their support and confidence are *R* times the expected *minsup* and minconf. GARPA algorithm [15] generalizes only if the descendents of an ancestor generate rules, and the rule of the ancestor has support value x% greater than the descendent that generate a rule with the biggest support among its siblings.

Table 1 compares the approaches mentioned in this section with NARFO algorithm. The presence of X in a cell indicates that the approach considers a specific feature.

Table 1. Comparison among approaches

Approaches	Generalized Association Rule	Redundancy Treatment	Fuzzy Association Rules	Generalization of non-frequent itemsets	Ontology
[10]	X	X			
[11]	X	X			
[15]	X	X			
[1]	X	X	X		
[8]	X		X		X
[4]	X				X
[9]	X	X			X
[2]			X		X
[13]	X	X		X	
NARFO	X	X	X	X	X

3 NARFO Algorithm

The NARFO algorithm extends and enhances the XSSDM algorithm [8] in some aspects. The first improvement is the generalization process, by including the generalization of infrequent itemsets, which is similar to the maximal frequent itemset technique cited in section 2. NARFO algorithm also performs a post-processing analysis, when rules are generated, by generalizing the rules that have all descendants

of the same immediate ancestor. Considering the fuzzy ontology of figure 1, if Tomato → Chicken and Cabbage → Chicken are generated rules, then the generalization of these rules is made, resulting in Vegetable → Chicken, since Tomato and Cabbage represent all descendants of Vegetable concept. Another relevant feature is the redundancy elimination, considering association rules like Apple~Tomato → Chicken and Apple → Chicken.

In figure 2, the steps of the algorithm are illustrated. The steps highlighted are the ones where generalization and redundancy elimination are performed. In the sections bellow, the steps of the NARFO algorithm are explained. Steps from 1 to 4, and step 6 are similar to the respective ones in XSSDM.

3.1 Data Scanning

This step identifies the items in the database generating itemsets of one size (1-itemst). These items have a straight correspondence to leaf-nodes of the fuzzy-ontology and, consequently, similarity relations to their siblings can be easily found. Considering the fuzzy ontology of figure 1, this step identified the following items: Apple, Kaki, Tomato, Cabbage, Chicken, Turkey and Sausage.

3.2 Identifying Similar Items

The similarity degree values between items are supplied by a fuzzy ontology, which specifies the semantics of the database contents. This step navigates through the fuzzy ontology structure to identify semantic similarity between items. If the similarity degree between items is greater than or equal to the *minsim* parameter explained in section 2, a semantic similarity association is found and this association is considered similar enough. A fuzzy association of size 2 is made by these pair of items found and are expressed by fuzzy items, where the symbol ~ indicates the similarity relation between items, for example, Kaki~Tomato.

After that, this step verifies the presence of similarity cycles as proposed in [16]. These are fuzzy associations of size greater than 2 that only exists if the items are, in pairs, sufficiently similar. The similarity cycles involve only leaf nodes of the ontology. The minimum size of a cycle is 3, and the maximum is the number of sibling leaf nodes.

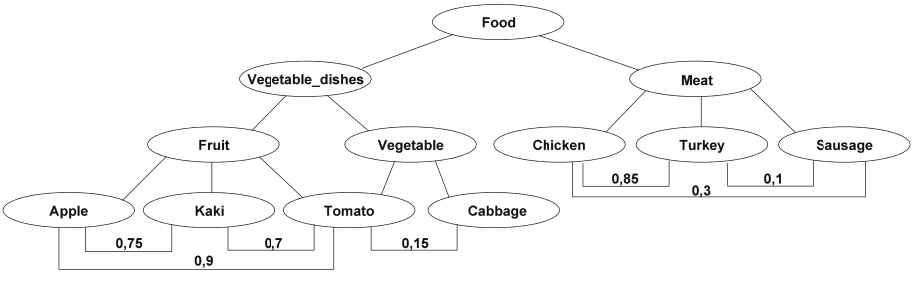


Fig. 1. Example of Fuzzy Ontology with fuzzy similarity degrees between items

In the end of this step, all fuzzy associations and similarity cycles are found and can be used to generate the rules.

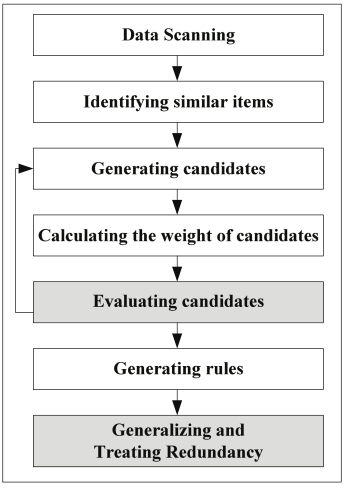


Fig. 2. Steps of NARFO algorithm

3.3 Generating Candidates

The generation of candidates in this algorithm is similar to Apriori algorithm. However, in NARFO algorithm, besides the items identified in the step described in section 3.1, fuzzy items, which represent fuzzy associations, are also added to generated candidates.

At the end of this step, we have all itemsets candidates of size k that are sent to the step described in section 3.4.

3.4 Calculating the Weight of Candidates

The weight of candidates is calculated based on the Fuzzy weight equation proposed in [16], if a candidate itemset is fuzzy. This equation was created due to the presence of fuzzy logic concepts.

The weight reflects the number of occurrences of an itemset in the database. In this step, the database is scanned, and each of its rows is confronted with the set of candidate itemsets, one after one. For each occurrence of a non-fuzzy candidate itemset in a row, its weight is incremented by 1. Otherwise, it is incremented by the fuzzy value, which is calculated based on the Fuzzy weight equation.

Finished this step, the itemsets candidates are ready to be used in next step.

3.5 Evaluating Candidates

This step is similar to the corresponding one in Apriori algorithm. The support of each itemset is evaluated. However, during the process, besides verifying if each candidate

itemset is frequent or not (has support greater than or equal *minsup*), the algorithm verifies the non-frequent itemsets, step which is not verified by neither Apriori nor XSSDM algorithms. If all descendents of an ancestor are non-frequents, but if the sum of the descendents' support is greater than or equal *minsup*, the NARFO algorithm generalizes the descendents to its ancestor and add the generalized itemset to the set of frequent itemsets. For example, considering the fuzzy ontology in figure 1, if ((Kaki), (Apple), (Tomato)) are non-frequent itemsets, the algorithm sum its supports. If the result is greater than or equal *minsup*, the generalization of the itemsets is done, and the ancestor Fruit is added to the set of frequent itemsets. This represents a meaningful and significant knowledge that was not extracted before by Apriori and other algorithms, since they do not consider the generalization of infrequent itemsets. It is one of the main contributions of this work. The pseudo-algorithm of this technique is showed in table 2.

Table 2. Pseudo-algorithm for generalizing non-frequent itemsets

```
1   for each ancestor
2      for each non-frequent itemset of size k
3         if itemset belongs to ancestor
4            sum support and increment counter
5         end if
6      end for
7      if counter is equal to the number of ancestor's child
8         add ancestor to frequent itemset // generalization
9      end if
10  end for
```

In table 2, lines from 2 to 6 count the number of descendents of an ancestor and sum their supports. Line 7 checks if all descendents of an ancestor are non-frequent itemsets, and add the ancestor (generalized itemset) to the set of frequent itemsets in line 8. This generalization can also be applied for itemsets with size greater than 1. For example, if ((Kaki, Turkey), (Apple, Turkey), (Tomato, Turkey)) are non-frequent itemsets, the generalization (Fruit, Turkey) is made since all descendents of Fruit composes the non-frequent itemsets and the other item, Turkey, belongs to all those non-frequent itemsets.

The support of an itemset is the weight divided by the number of transactions in the database. If the item set is fuzzy, the algorithm divides the fuzzy weight to the total of transactions. If a candidate itemset is frequent, its support is greater than or equal *minsup*.

In the end of this step, we have all frequents itemsets of size k, where k is the size of the itemset. The algorithm returns to the step explained in section 3.3 to generate the candidate itemsets of size k + 1. If k is equal to the number of domains, NARFO algorithm goes to step 3.6.

3.6 Generating Rules

In this step, the association rules are generated for each itemset of the set of frequent itemsets. All possibilities of antecedents and consequents are generated. The rules that have the confidence value greater than or equal *minconf* are considered strong.

Remember that, the confidence of a rule is given by the support of the rule divided by the support of the antecedent.

Then, the algorithm verifies these rules in order to check if a fuzzy item can be generalized. A fuzzy item can be generalized if all descendents of an ancestor are contained in the fuzzy item. If the rule Tomato~Cabbage → Turkey exists, then the rule Vegetable → Turkey is generated, since Vegetable comprises Tomato and Cabbage as descendant concepts.

After that, all the rules generated are sent to the next step to verify other possible generalization and redundant rules.

3.7 Generalizing and Treating Redundancy

After all rules have been generated by the previous step, they receive generalizing and redundancy treatment. The corresponding pseudo-algorithm of these treatments is showed in table 3.

For all rules, the algorithm verifies if the antecedent / consequent of each rule can be generalized (lines 1-6 in table 3). For example, if the algorithm generated the rules Kaki → Sausage and Apple~Tomato → Sausage, then the algorithm does the generalization to Fruit → Sausage because all descendants of Fruit occurred in rules (including fuzzy items) with the Sausage concept as consequent. The generalization process happens not only if all descendents of an ancestor are in a fuzzy item, but also if all descendents of an ancestor are in different rules that have the same corresponding antecedent or consequent. This can happen at the antecedent or consequent of a rule.

In the redundancy treatment, the algorithm deals with two redundancy issues. The first one is when a rule is sub-rule of another one (lines from 7 to 11 in table 3). A sub-rule r_1 is a rule that has the same items in antecedent and consequent considering another rule r_2, except that r_1 has at least one item that is descendent of an item in r_2, in the same side of the rule (antecedent or consequent). Then, r_1 is eliminated. For example, the rule Kaki → Sausage is a sub-rule of Fruit → Sausage as Kaki is descendent of the ancestor Fruit. Then, the rule Kaki → Sausage is eliminated.

Table 3. Pseudo-algorithm of the generalization and redundancy treatment

```
1   for each rule
2      if rule can be generalized //antecedent or consequent
3         generalize rule
4      end if
5      add rule to v // v is a vector of rules
6   end for
7   for each rule from v
8      if rule is not a sub-rule
9         add to v1 // v1 is auxiliary vector of rules
10     end if
11  end for
12  for each rule from v1
13     if rule is not a fuzzy sub-rule
14        add to v2 // v2 is auxiliary vector of rules
15     end if
16  end for
```

The other redundancy treatment is the fuzzy redundancy (lines 12-16 in table 3). This kind of redundancy occurs when a rule is fuzzy sub-rule of another rule that contain a fuzzy item. A fuzzy sub-rule r_1 is a rule that has the same items, in antecedent and consequent of another rule r_2, except that r_1 has at least one item that is contained in a fuzzy item, in the same side, in r_2. For example, if the algorithm generates the rules Apple → Sausage and Apple~Tomato → Sausage, the first one is pruned. Only Apple~Tomato → Sausage is showed to the user. When this case happens, the resultant rule is showed to the user in this format: Apple~Tomato → Sausage with sup: 0.3 and conf: 0.45 (Item 'Apple' has more relevance). The phrase between the parentheses is showed in order to highlight the item in the fuzzy item that has more relevance, once there was a redundant fuzzy sub-rule with this relevant item.

4 Experiments

In this section we show some experiments performed to validate the NARFO algorithm.

In these experiments, we have considered data from the Brazilian Demographic Census 2000, provided by IBGE (Brazilian Institute of Geography and Statistics). Two databases containing information about demographic characteristics of the Brazilian population were analyzed: *IBGE1*, which contains information about *Years of study, Race or ethnicity and Sex*; and *IBGE2*, containing relations between *Race or ethnicity and Living (urban* or *rural)*.

After analyzing the data and domain of demographic characteristics, the fuzzy ontology of figure 3 was created. The ontology was modeled in OWL (Web Ontology Language) and Jena Framework [17] was used to allow navigation through ontology concepts and relations, making NARFO algorithm able to obtain similar items and corresponding similar degrees.

The tests were done with constant *minconf* and *minsim*, whose values are, respectively, 0.2 and 0.1. The *minsup* value varies from 0.05 to 0.5, increasing it by 0.05.

Two tests (*Test A* and *Test B*) were done and compared with the XSSDM algorithm, regarding *IBGE1* database. *Test A* considers the generalization and the redundancy treatment described in section 3.7, and compare the number of association

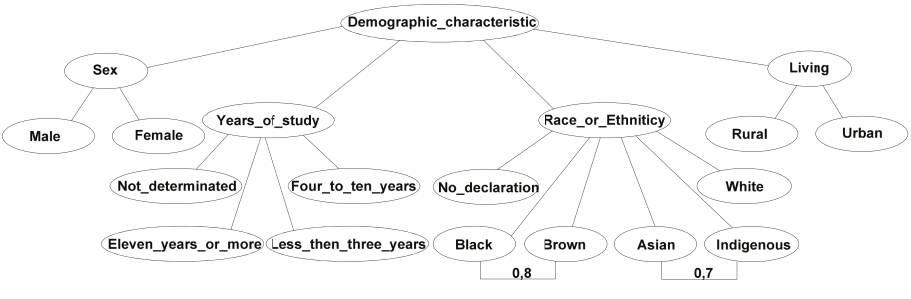


Fig. 3. Fuzzy Ontology of Demographic Characteristics

rules generated in NARFO and XSSDM algorithms. The aim of *Test A*, without the generalization of non-frequent itemsets, is to confirm that NARFO algorithm extracts fewer association rules in comparison to XSSDM. With *minsup* = 0.05, NARFO generates 46 rules against 91 of XSSDM (49.45% of reduction). With *minsup* = 0.25, 4 rules were generated by our algorithm against 6 of XSSDM, and both algorithms do not generate any rule with *minsup* = 0.3. The results are illustrated in Figure 4.

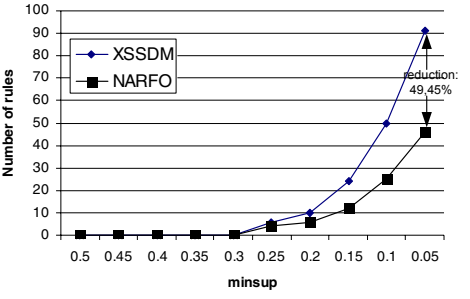


Fig. 4. NARFO only with generalization and redundancy treatment (*Test A*)

Test B compares the number of association rules extracted by NARFO considering all the techniques described in this paper, including the generalization of non-frequent itemsets described in section 3.5 and XSSDM. The results of *Test B* are illustrated by Figures 5 and 6. Figure 5 shows that NARFO algorithm generates fewer rules than XSSDM algorithm when considering low support values, because NARFO is able to eliminate redundant rules.

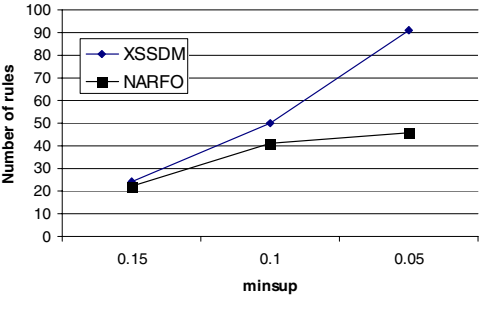


Fig. 5. Full NARFO with low *minsup* values (*Test B*)

In Figure 6, NARFO generates more rules than XSSDM when the *minsup* is between 0.2 and 0.5. This happens because NARFO algorithm generalizes descendents of an ancestor if all of them are non-frequent, resulting in relevant rules that represent new and meaningful knowledge that was not found by the XSSDM algorithm, representing an important contribution of this work.

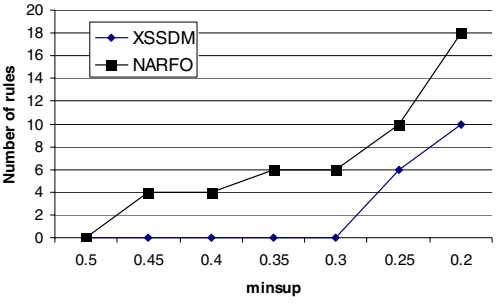


Fig. 6. Full NARFO with high *minsup* values (*Test B*)

We have also tested both algorithms with *IBGE2* database regarding information on *Race or ethnicity and Living (urban or rural)*. The results are shown in Figure 7. For *IBGE2* database, NARFO reduces the number of generated rules in 63.63% (4 rules against 11 of XSSDM), when support is below 0.3. Therefore, the generalization and redundancy treatment performed by NARFO is able to prune irrelevant rules that are generally obtained when considering low *minsup* values.

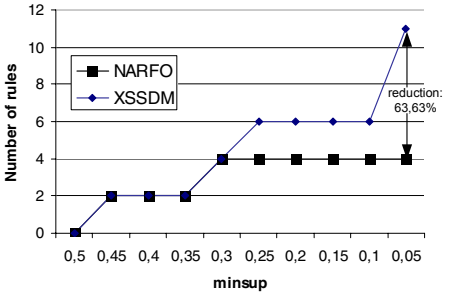


Fig. 7. Full NARFO tested for *IBGE2* database

To sum up, we could observe two distinct situations from the results. When considering low *minsup* values, NARFO reduces the number of rules because it performs generalization and removes redundant association rules without lost in semantics, comparing to the XSSDM algorithm. On the other hand, when *minsup* is increased to high values, NARFO algorithm produces the same amount or more meaningful rules, due to the generalization of non-frequent itemsets. In this case, the additional rules express relevant knowledge, since they are based on the semantic concepts and relationships of the fuzzy ontology.

5 Conclusions and Future Work

This paper proposes a new algorithm, called NARFO algorithm, for mining non-redundant and generalized association rules based on fuzzy ontologies. Our algorithm

does an efficient generalization and redundancy treatment without lost of information. Experiments have demonstrated that irrelevant rules are pruned, resulting in a smaller amount of redundant rules.

Furthermore, during the evaluation of candidates, NARFO performs generalization when all descendents of an ancestor are non-frequent itemsets, which provides meaningful knowledge, as stated by the tests. Hence, it is possible to obtain more relevant rules based on semantic information of fuzzy ontologies, therefore enhancing the knowledge discovery process.

We are implementing some improvements of the NARFO algorithm. For example, we intend to include a new parameter for generalization. Considering this, if X% of descendents are included in rules, the generalization is done. The algorithm will also show the descendents that do not generate a rule, in order to avoid showing mistaken information for users.

Acknowledgements. This work has been supported by the following Brazilian research agencies: CAPES, CNPq, FAPESP, FINEP and INEP. The first two authors also thank the support of the Web-PIDE Project in the context of the Observatory of the Education of the Brazilian Government.

References

1. Chen, G., Wei, Q., Kerre, E.E.: Fuzzy Data Mining: Discovery of Fuzzy Generalized Association Rules. In: Bordogna, G., Pasi, G. (eds.) Recent Issues on Fuzzy Databases, pp. 45–66. Physica-Verlag, Würzburg (2000)
2. Farzanyar, Z., Kangavari, M., Hashemi, S.: A New Algorithm for Mining Fuzzy Association Rules in the Large Databases Based on Ontology. In: Sixth IEEE International Conference on Data Mining – Workshops, Hong Kong, China, December 18-22 (2006)
3. Srikant, R., Agrawal, R.: Mining Generalized Association Rules. In: Proceedings of the International Conference of Very Large Data Bases, Zurich, Switzerland, September 11-15 (1995)
4. Hou, X., Gu, J., Shen, X., Yan, W.: Application of Data Mining in Fault Diagnosis Based on Ontology. In: 3th Third International Conference on Information Technology and Applications, Sydney, Australia, July 4-7 (2005)
5. Zadeh, L.: Similarity Relations and Fuzzy Orderings. In: Yager, R.R., Ovchinnikov, S., Tong, R.M., Nguyen, H.T. (eds.) Fuzzy Sets and Applications: Select Papers by L. A. Zadeh, pp. 81–104. Wiley Interscience, New York (1987a)
6. Zadeh, L.: Fuzzy Sets. In: Yager, R.R., Ovchinnikov, S., Tong, R.M., Nguyen, H.T. (eds.) Fuzzy sets and applications: Selected Papers by L.A. Zadeh, pp. 29–44. Wiley-Interscience, New York (1987b)
7. Chen, X., Zhou, X., Scherl, R.B., Geller, J.: Using an Interesting Ontology for Improved Support in Rule Mining. In: 5th InternationalConference on Data Warehousing and Knowledge Discovery, Prague, Czech Republic, September 3-5 (2003)
8. Escovar, E.L.G., Yaguinuma, C.A., Biajiz, M.: Using Fuzzy Ontologies to Extend Semantically Similar Data Mining. In: 21st Brazilian Symposium of Databases, Florianópolis, Brazil, October 16-20 (2006)
9. Brisson, L., Collard, M., Pasquier, N.: Improving Knowledge Discovery Process Using Ontologies. In: International Workshop on Mining Complex Data, Houston, USA, November 27-30 (2005)

10. Han, J., Fu, Y.: Mining Multiple-Level Association Rules in Large Databases. IEEE Transactions on Knowledge and Data Engeneering 11(5), 798–805 (1999)
11. Sriphaew, K., Theeramunkong, T.: Fast algorithms for mining generalized frequent patterns of generalized association rules. IEICE Transactions on Information and Systems E87-D(3), 761–770 (2004)
12. Pasquier, N., Bastide, Y., Taouil, R., Lakhal, L.: Discovering frequent closed itemsets for association rules. In: Proceedings of the International Conference on Database Theory, Jerusalém, Israel, January 10-12 (1999)
13. Kunkle, D., Zhang, D.H., Cooperman, G.: Mining Generalized Frequent Itemsets and Generalized Association Rules Without Redundancy. Journal of Computer Science and Technology 23(1), 77–102 (2008)
14. Bayardo, J.R.J.: Efficiently mining long patterns from databases. In: Proceedings of the 1998 Annual Conference on Management of Data, Seattle, USA, June 2-4 (1998)
15. Oliveira, V.C., Rezende, S.O., Castro, M.: Evaluating Generalized Association Rules Through Objective Measures. In: Proceedings of 25th International Multi-Conference on Artificial Intelligence and Applications, Innsbruck, Austria, February 12-14 (2007)
16. Escovar, E.L.G., Biajiz, M., Vieira, M.T.P.: SSDM: A Semantically Similar Data Mining Algorithm. In: 20th Brazilian Symposium of Databases, Uberlândia, Brazil, October 3-7 (2005)
17. Carrol, J.J., Dickinson, I., Dollin, C., Reynolds, D., Seaborn, A., Wilkinson, K.: Jena: implementing the semantic web recommendations. In: International World Wide Web Conference, New York, USA, May 19-21 (2004)

Automated Construction of Process Goal Trees from EPC-Models to Facilitate Extraction of Process Patterns

Andreas Bögl[1], Michael Schrefl[1], Gustav Pomberger[2], and Norbert Weber[3]

[1] Department of Business Informatics, Data & Knowledge Engineering
Johannes Kepler University Linz, Altenberger Straße 69, 4040 Linz, Austria
{boegl,schrefl}@dke.uni-linz.ac.at
[2] Department of Business Informatics, Software Engineering
Johannes Kepler University Linz, Altenberger Straße 69, 4040 Linz, Austria
gustav.pomberger@jku.at
[3] Siemens AG, Corporate Technology-SE 3, Otto-Hahn-Ring 6, Munich, Germany
norbert.weber@siemens.com

Abstract. A system that enables reuse of process solutions should be able to retrieve "common" or "best practice" pattern solutions (common modelling practices) from existing process descriptions for a certain business goal. A manual extraction of common modelling practices is labour-intensive, tedious and cumbersome. This paper presents an approach for an automated extraction of process goals from Event-driven Process Chains (EPC) and its annotation to EPC functions and events. In order to facilitate goal reasoning for the identification of common modelling practices an algorithm (G_{Tree}-Construction) is proposed that constructs a hierarchical goal tree.

Keywords: Extraction of Process Goals, Semantic EPC Models, Process Patterns, Common Modelling Practices.

1 Introduction

Organizations have to cope with a steadily changing environment *„driven by such pressures as customer expectations, new technologies and growing global competition"* [2]. They have to pay much attention to organize and optimize their business processes. Therefore, business process management in general and the discipline of business process modelling in specific are relevant for such organizations. In this work, we assume process descriptions in terms of Event-driven Process Chains (EPC) [11] since this modelling language has gained a broad acceptance and popularity both in research and in practice. EPC models describe business processes on a management and organizational level. The model element structure is expressed in terms of the meta language constructs functions, events and connectors (and, xor, or). Natural language expressions describe the implicit meaning of functions and events.

During the design or adaptation of business processes, process modellers want to have access to proven pattern solutions (common modelling practices) implicitly modelled in existing EPC models. Therefore, such process knowledge should be automatically extracted from given process descriptions. Common modelling

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 427–442, 2009.
© Springer-Verlag Berlin Heidelberg 2009

practices can be described in terms of process patterns [5]. A process pattern represents a *"common"* or *"best practice solution"* to solve a particular problem or to meet a certain process goal in a certain context. Therefore, it might assist process modellers for constructing high quality process solutions.

The work presented in this paper relies on research activities being part of the BPI (Business Process Improvement) project[1]. The project addresses the issue *"How can process patterns be automatically extracted from given EPC models in engineering domains?"* The automated extraction of process patterns from given process models yields several advantages. Using a large set of specific models offers a detailed insight into the common and best practices of a domain. The frequency of occurrence enables an objective measure to evaluate candidates for common modelling practices.

A problem solved by a process pattern can be interpreted in the sense of how to achieve a defined process goal. A goal *"represents the purpose or the outcome that the business as a whole is trying to achieve. Goals control the behaviour of the business and show the desired states of some resource in the business" [13].*

Usually the identification and design of process models is derived from the goals or objectives of an enterprise, because the processes are a means to achieve the goals. However, when using the EPC modelling language for process descriptions, process goals are not explicitly modelled in practice. Hence, the idea is to perform a retroactive extraction of goals from given EPC models and to annotate EPC functions with goals. According to [12], *"the aim of goal annotation is to pragmatically facilitate recognizing process knowledge conveyed by heterogeneous process models based on the enriched intentional semantics of processes".*

This paper addresses the following issues when extracting process patterns from given EPC models:

(1) How Process Goals are Extracted from and Annotated to EPC Models? Since we advocate a state based description of process goals, EPC events provide potential candidates for extracting process goals. In practice, many EPC events are not modelled explicitly, especially trivial events since they cause an undesired growing of model size. Further, goal satisfaction also depends on constraints which additionally define conditions that must be fulfilled by a pattern solution for goal satisfaction.

(2) How to Identify Relationships between Process Goals from Goal Annotated EPC Models? Process goals are not isolated items within a goal space. Usually, goals are organized in terms of a taxonomy which decomposes goals into a set of subgoals each has assigned at least one pattern solution. The decomposition of goals into subgoals also has to consider relationships which reflect goal dependencies between arranged subgoals. For instance, goal satisfaction may require to meet subgoals sequentially or two subgoals represent alternatives to each other.

Such an organization of process goals yields several advantages. Following, we highlight two practical benefits exemplarily. Hierarchical goal trees support to find matching process patterns even though process solutions are modelled on different levels of detail.

Fig. 1 illustrates a simplified example that shows two EPC models each providing a process solution to specify requirements (e.g. Software Requirements). EPC model

[1] The project was funded by Siemens AG, Corporate Technology – SE 3, Munich.

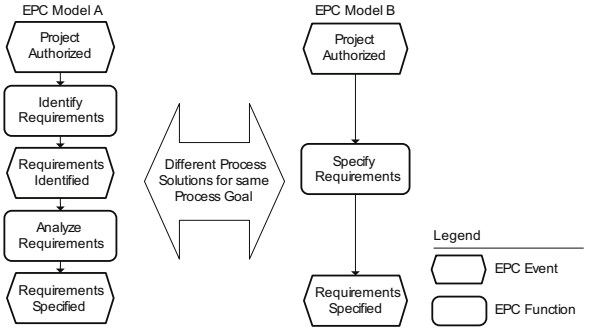


Fig. 1. Example for different Process Solutions achieving a same Process Goal

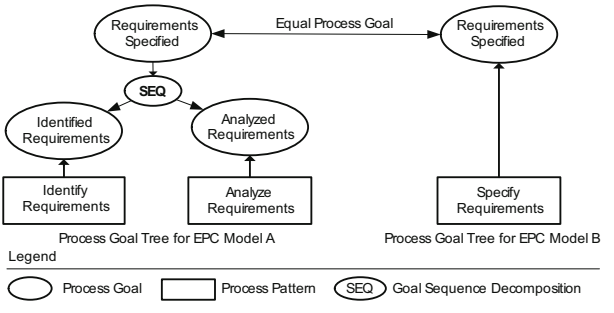


Fig. 2. Example for Process Goal Trees, derived from EPC Models in Fig. 1

A provides a more detailed process solution than EPC model *B*, since it comprises the two EPC functions *"Identify Requirements"* and *"Analyze Requirements"* to specify requirements. EPC model *B* simply models one EPC function named *"Specify Requirements"*. Consequently, these two process solutions achieve an equal process goal however modelled on a different level of detail. The goal trees depicted in Fig. 2 are derived from the EPC Models of Fig 1.

The goal tree for EPC model *A* consists of a goal sequence decomposition that decomposes the process goal *"Requirements Specified"* into the two subgoals *"Identified Requirements"* and *"Analyzed Requirements"* whereas the goal tree for EPC model *B* only comprises the process goal *"Requirements Specified"*. Since the root goals of each goal tree refer to an equal process goal assigned process patterns represent variants for achieving the root goal *"Requirements Specified"*.

Process patterns are interrelated to specify larger coherent structures (composite process patterns). Such relationships between process patterns can be derived from dependencies between process goals. The composite process pattern illustrated in Fig. 3 is derived from the process goal trees illustrated in Fig. 2. The patterns *"Identify Requirements"* and *"Analyze Requirements"* are part of sequentially decomposed subgoals. Thus, these two patterns are interrelated by a <<*hasSuccessor*>> relationship.

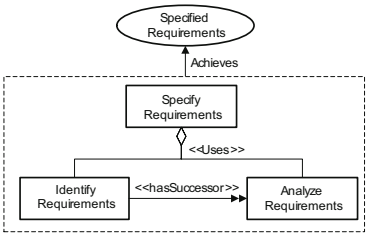


Fig. 3. Example for Composite Process Pattern

A <<*hasSuccessor*>> relationship connects two process patterns sequentially. Since the process pattern *"Specify Requirements"* provides a more comprehensive process solution, it *uses* the two process patterns *"Identify Requirements"* and *"Analyze Requirements"*.

Same or similar process goals can be met by using same EPC functions and events however expressing a different control flow. One solution may model functions as a sequence, another one arranges them as a parallelization, for instance. Extracted process patterns have to consider such different relationships between process solutions.

Let us consider Fig. 4 that illustrates two process solutions each comprising the same EPC functions and events. The first EPC solution models functions and events sequentially, the second one considers that the two functions *"Identify Requirements"* and *"Analyze Requirements"* are executed parallel to each other (Fig. 4(a)). This

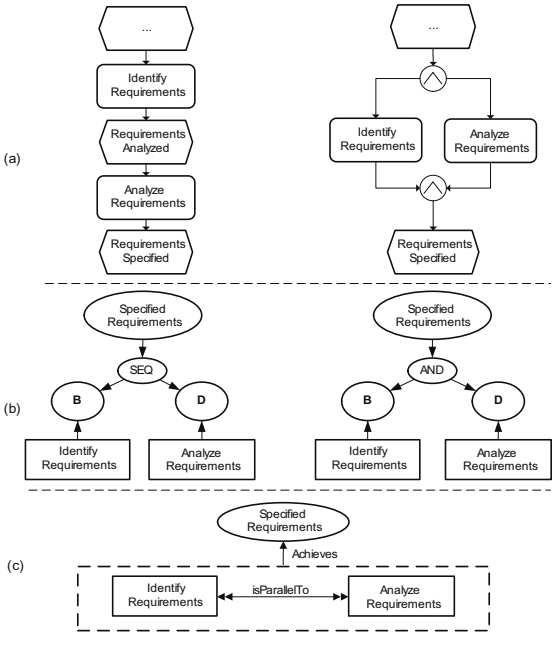


Fig. 4. Example for different Control Flows of equal EPC Functions and Events

entails that the first goal tree considers a goal *"SEQUENCE"* decomposition whereas the second one expresses the parallelization by an *"AND"* decomposition (Fig 4(b)). This implies that the two process patterns are interrelated by the two semantic pattern relationship <<*isParallelto*>>. This relationship means that two process patterns can be used independently from each other or sequentially (Fig 4(c)).

In this paper, we propose an approach that constructs a hierarchical goal tree that meets above mentioned requirements for extracting process patterns from given EPC models. To enable this, we perform a structural transformation of EPC models to achieve a tree-based goal description.

The rest of this paper is organized as follows: The upcoming section introduces structures and semantically annotated EPC models that represent the input for the construction of process goal trees. Section 3 describes how process goals are organized in terms of a hierarchical process goal tree. In Section 4 we introduce the *algorithm G-Tree Construction*. Section 5 elaborates related work in the area of process goal modelling. The paper concludes with a summary of practical experiences.

2 Structures and Semantically Annotated EPC Models

An EPC model is a directed and connected graph whose nodes are events, functions and logical connectors which are connected by control flow arcs. Natural language expressions describe the implicit meaning of EPC functions and events. Such expressions follow naming conventions or standards that represent guidelines for naming EPC functions/events [16]. If a naming convention is used, the meaning of a lexical term is clear. For instance, a task is always expressed by an active verb whereas a process object is of word type noun. Functions represent time and cost consuming elements for performing a task on a process object, e.g. the task *"Define"* is performed on the process object *"Software Requirements"* to achieve a certain process goal. Events represent states for process objects in the execution of functions and control the further process flow. State information for process objects may either have a local or global scope. A local scope refers to state information directly produced by an EPC function whereas a global scope refers to a partial or to a whole process model. For example, if the event *"Software Requirements Identified"* directly succeeds to the function *"Identifiy Software Requirements"* then this event indicates a trivial event. An end event (e.g. *"Project Planned"*) of a process always represent a non-trivial event since it refers to the whole underlying process, for instance. Functions can be triggered by more than one event or can produce more than one state. In order to model such requirements it is necessary to capture the business logic of a business process by using logical operators. Logical operators enable to model a complex non-linear control flow of events and functions. They allow to model parallel ("and") or alternative ("xor", "or") paths.

2.1 Structured EPC Models

The extraction of process goals from given EPC models and its decomposition in terms of a hierarchical tree structure assumes structured EPC models. Structured EPC models follow well-established style rules, developed by [8]. Following, we briefly

summarize these modelling practices. Fig. 5(a-d) illustrates well-structured workflow constructs. These constructs demand that for each join connector exists a corresponding split connector of the same type. Rule 2 of Fig. 5(e) means that a control flow may consist of another well-structured workflow construct what is called nested structure. Fig. 5(f-g) specify how exit points from a control flow are modelled. According to [8], Rule 3 *"allows to jump out of a split-join construct, but it is not allowed to jump into a split-join construct"*. It is important to note that the split does not have to be a XOR-split. In this work, split connectors that represent an exit point or jump out are called *exit split connectors*.

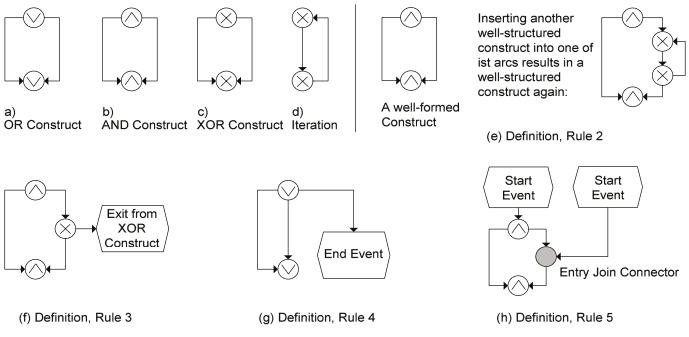


Fig. 5. Workflow Constructs and Definition of Rules

The proposed style rules do not support modelling multiple start events whose control flow jumps into a split-join construct. Therefore, Rule 5 (Fig. 5(h)) denotes an additional rule not explicitly specified by [8]. It specifies an *entry point* within a split-join construct that is realized by an *entry join connector*.

2.2 Automated Semantic Annotation

An automated extraction of process patterns from EPC models requires to perform a semantic analysis of EPC functions and events. The semantic analysis faces the problem that an essential part of the EPC semantics is bound to natural language expressions in functions and events with undefined process semantics. Consequently, the meaning is not machine-interpretable. To tackle this problem, EPC functions/events are annotated with semantic linkages to a reference ontology that captures the meaning of used vocabulary in terms of process objects, tasks and state information. This annotation empowers computer systems to process the meaning of EPC functions/events. Fig. 6 depicts an example for a semantically annotated EPC model. The right part illustrates a snapshot of the reference ontology that captures instances of the concepts *Process Object (PO)* (e.g. *Requirements*), *Task* (e.g. *Identify*) and *State* (e.g. *Identified*) and its semantic relationships (e.g. *isPartOf*). A semantic linkage from EPC functions and events to reference ontology instances ensures a unique meaning of used process vocabulary. In [3], a comprehensive approach that performs an annotation of EPC models and that populates a knowledge base automatically, is presented.

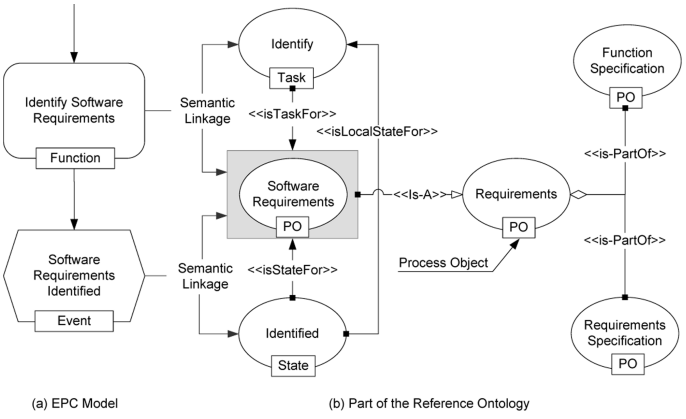


Fig. 6. Example for Semantically Annotated EPC Model

Further, an ontology-based annotation of EPC functions and events enable to identify semantic dependencies between process objects associated tasks and state information, for instance. Reconsider Fig. 6, the process object "*Software Requirements*" is a specialisation of the more abstract process object "*Requirements*". Such relationships between knowledgebase instances enable to match process goals at different levels of abstraction, for instance.

3 Process Goal Trees

The aim of a process pattern is to provide a process solution to meet a business goal in a certain situation. This work advocates a state based description of process goals according to [13]. For example, the process *goal "Identified Requirements for Software"* refers to the process object *"Requirements for Software"* with the state *"Identified"*. In other words, a process pattern provides a process solution for achieving the state *"Identified"* for the process object *"Requirements for Software"*. A process goal tree results from a decomposition of process goals into subgoals through and/or/xor graph structures borrowed from problem reduction techniques in artificial intelligence. A process goal tree G_{Tree} is a bipartite graph *(N,V)* where $N = (G \cup D)$, $(G \cap D) = \emptyset$ is a set of nodes, G is a set of goals and D is a set of decompositions *{sequence, and, or, xor}* over G and V is binary relation of decomposition arcs $V \subseteq (G \times D) \cup (D \times G)$ such that V specifies a tree with a goal node $r \in G$ as its root.

Process goal satisfaction depends on constraints assigned to process goals. A constraint $c \in C = (C^R \cup C^L)$ denotes a precondition that refers to a process object with certain state information. Constraints are classified into *global constraints* (C^R) and *local constraints* (C^L). *Global constraints* are assigned to the process goal that is satisfied by the whole underlying process whereas a *local constraint* refers to a decomposed subgoal. A constraint C is a quadruple *(P,S,V,T)* where P is a process object, S is a state associated with the process object, T = *{boolean, discrecte,*

numeric} is a type for a value assigned to *V*. (e.g. C_1 = *("Development Risk","Low","True","boolean")*). Constraints can be composed to larger structures coupled by the prepositional expressions *"and"*, *"or"*, *"xor"*.

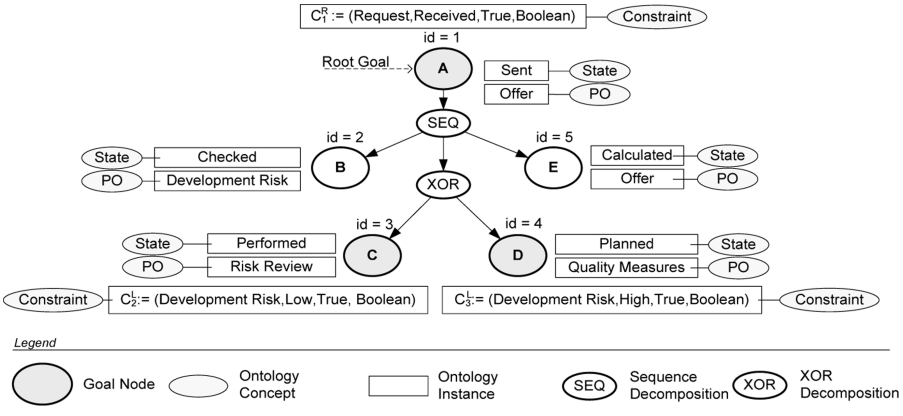


Fig. 7. Example for Constraints of Process Goals

Fig. 7 illustrates an example for a process goal tree. The root goal *"Sent Offer"* requires that a customer request must have the state "*Received = True"*. This is expressed by the global constraint C_1^R = *("Request", "Received", "True", "Boolean")*). In order to satisfy the goal *"Performed Risk Review"*, the goal *"Checked Development Risk"* must be satisfied. Further, a satisfaction of goal *C* also requires that the development risk must be low that is expressed by the *local constraint* C_2^L =*("Development Risk", "Low", "True", "Boolean")*. The process goals *C* and *D* indicate goal variants, the satisfaction of such a variant depends on the defined constraints.

A process goal $g \in G = (G^C \cup G^E)$ is either an elementary goal ($g \in G^E$) or a non-elementary goal ($g \in G^C$). Elementary goals are not further decomposed into subgoals ($G^E = \{g \in G | (|g_{out}| = 0)\}$) whereas non-elementary goals are decomposed into a set of subgoals. A process goal is a structure $G = (id, S, P, C)$ consisting of a function $id : G \rightarrow Integer$ that is a unique identifier for a process goal *G*, *S* is state for process object *P* and *C* a set of constraints associated with a goal.

A decomposition of a goal into a set of subgoals implies a satisfaction of subgoals to meet a parent goal by considering the following decompositions. A *sequence* decomposition decomposes a goal into an ordered tupel of subgoals whereas each must be met in a certain sequence. An *and* decomposition expresses an arbitrary satisfaction order of all subgoals associated with a parent goal; a *xor* decomposes a goal into a disjoint set of subgoal variants; an *or* implies a satisfaction of either one, multiple or all subgoals to comply with the parent goal.

4 Construction of Process Goal Trees

The construction of process goal trees comprises the following three steps: (1) splitting EPC models with more than one end event into multiple EPC models each having exactly one end event, (2) annotation of EPC functions and events with process goals and (3) organizing annotated process goals in terms of a goal tree.

1. Splitting of EPC Models. As already mentioned an end event represents the root process goal that is decomposed into a set of sub goals. Hence, if an EPC model comprises multiple end events it must be split into partial EPC models each having exactly one end event from which the top level goal is extracted from (see Fig 8).

The split-up of EPC models is based on *exit split connectors* as introduced in section 2.1. *Exit split connectors* define a process path for an additional end event.

2. Annotation with Process Goals. EPC functions and events are annotated with process goals and associated constraints. For goal annotation, the distinction between trivial events and non-trivial events plays a vital role. Trivial events always capture elementary process goals (G^E) whereas non-trivial events represent process goals that are further decomposed into sub goals. Further, extracted process goals from functions always denote elementary process goals since the extracted process goal refers to the process object modelled in a function and the state information for this process object is a local state that the task of a function refers to. For example, the function *"Identify Requirements"* automatically implies the trivial event *"Requirements Identified"*. Generally, elementary goals (G^E) are extracted from semantically annotated EPC functions and non-elementary process goals (G^C) from semantically annotated non-trivial events. Global constraints associated with goals are extracted from start events, local constraints associated with goals from events that capture the result of a decision function.

Fig 9 illustrates an example for annotated process goals to EPC functions and events. Root goal G_{Root} is extracted from end event E_{End-1}. Goals G_{F-1}, G_{F-2}, G_{F-3} and G_{F-4} are elementary goals extracted from functions F_1, F_2, F_3 and F_4. The root goal constraints C_1 and C_2 are derived from start events $E_{Start-1}$ and $E_{Start-2}$. Local constraints C_3 and C_4 result from the events E_1 and E_3 expressing that the function F_1 produces state information that results from decision function F_1 (e.g. "*Check Development Risk*").

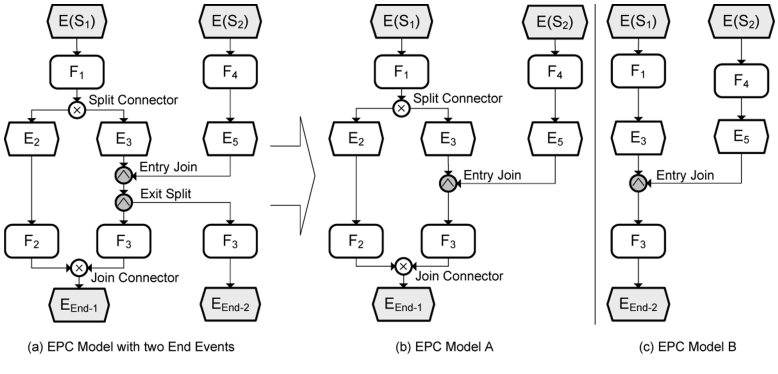


Fig. 8. Example for Model Split

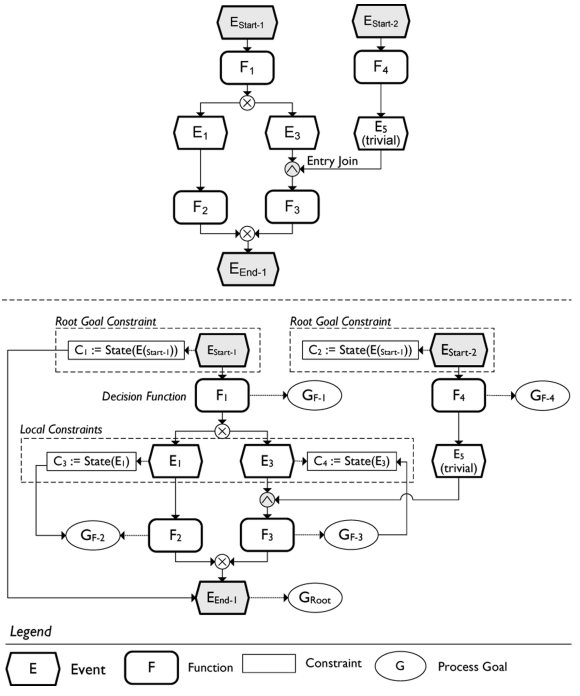


Fig. 9. Illustration for Annotation of Process Goals to EPC Functions/Events

3. Decoupling of Goal Structure from EPC Model: The process goal tree decouples the process goal structure from the goal annotated EPC structure by organizing annotated process goals in terms of a hierarchical tree. Input for the hierarchical tree construction is a semantically annotated EPC model with process goals that result from the previous step; output is a process goal tree (G_{Tree}) for that EPC model. Following, we present the algorithm G_{Tree}-*Construction* that constructs a process goal tree for a given EPC model annotated with process goals. Fig 10 illustrates the steps performed by the G_{Tree}-*Construction* algorithm to construct a goal tree for the example depicted in Fig. 9. G_{Tree} construction constitutes the main method. Since an EPC model may have one or several entry joins (c.f. Fig. 5) the goal tree construction invokes a single goal tree construction each constructing a goal tree for a partial process chain ($sEPC_{Split}$). In Fig. 9, a single goal tree is constructed for the process chain $E(S_2) \rightarrow F_4 \rightarrow E_5$ since the control flow concludes by an *"AND"* entryJoin.

The algorithm starts with a semantically annotated end event of an EPC model a goal tree is constructed for. Step 1 initializes an *entryJoinQueue* that captures *entryItems* and *entryJoins* followed by an initialization of the G_{Tree} (goal tree). If the model item denotes an end event, then a root goal added with a sequence node is initialized. Step 2 invokes the method *Single-GoalTree-Construction* that iterates recursively through all modell items and inserts - in dependency of the model item type - goal nodes or decomposition nodes to the goal tree (Step 3). If an *entryJoin* is detected then the current goal node and the entry item (last node that concludes with entry Join) are added to the *entryJoinQueue* (Step 4). If all model items have been processed the algorithm returns the root node (t) of the constructed G_{Tree} (Steps 5 – 9).

Algorithm 1: G_{TREE}-CONSTRUCTION

Input: *goal annotated modelItem*
Output: G_{Tree}*=(N,V) for goal annotated EPC model*

```
1  entryJoinQueue • ∅ {Queue that captures entryItems for an
entryJoin}
```
{Initialization of G_{Tree}, if modelItem is the end event, then construct root goal and add sequence node, else construct sequence node and add goal of modelItem as child node}
```
2  currentNode • InitGTree(GTree,modelItem)
3  G_Tree ← SINGLE-GOALTREE-CONSTRUCTION (currentNode, modelItem,
   entryJoinQueue)
```
{Construct single goal trees for each entryJoin}
```
4  while (entryJoinQueue not empty ) do
5  goalItem, entryItem ← Enqueue(entryJoinQueue)
6  G_Tree ← G_TREE-CONSTRUCTION (modelItem)
7  Merge resulting G_Tree with goalItem {Merge single goal trees to one goal tree}
8  return currentNode
```

SINGLE-GOALTREE-CONSTRUCTION (*currentNode, modelItem, entryJoinQueue*)

Input: *currentNode*, *modelItem, entryJoinQueue*
Output: partial G_{Tree} for $sEPC_{Split}$ *(partial process* chain)

```
   {break condition for recursion}
9  if ( isNil(modelItem) or isVisited(modelItem) or not
   allSuccessorsVisited(modelItem))
   then
   {not all paths have been processed, set new sequence node for next path}
10 if( isSplitConnector( modelItem ) then
11 set currentNode to parentNode of currentNode
12 insertSequenceNode( currentNode ) {adds Sequence Node to current Node}
13 return {Trace back to Join Connector}
   {all paths have been processed trace back to connector's parent node}
14 if( isSplitConnector( modelItem ) then
15 set currentNode to parentNode of parent of currentNode
16 else if (isJoinConnector( modelItem ) then {insert decomposition node
   in tree}
17 insertDecompositionNode( currentNode, modelItem )
18 insertSequenceNode( currentNode )
19 else if (isEntryJoinConnector(modelItem ) then
{entryJoin may have more than one entryItem}
20 insertDecompositionNode(currentNode)
21 for all( entryItem of modelItem ) do
22    insertSequenceNode (currentNode)
23    Enqueue currentNode and entryItem of entryJoin to
      entryJoinQueue
24    set currentNode to parentNode of parent of currentNode
25 else
26    insertGoalNode( currentNode, modelItem )
27 set visited flag of modelItem to true
28 for all (predecessors of modelItem) do
29    if ( not isEntryConnector( modelItem ) then
30    G_Tree ← SINGLE-GOALTREE-CONSTRUCTION (currentNode,
      modelItem,
      entryJoinStack)
31 set currentNode to root node of constructed G_Tree
32 return currentNode
```

If the *entryJoinQueue* is not empty then the G_{Tree}-Construction is invoked recursively (Step 10) in order to construct a G_{Tree} for each partial process chain concluding with an *entryJoinConnector*. Step 11 initializes a new G_{Tree}. Since model item E_5 is not an end event a sequence node as child node is added. Since item E_5 indicates a trivial event method *insertDecompositionNode(...)* neglects this item. Step 12 inserts goal items associated to remaining model items. If all single G_{Trees} for process chains have been constructed, they are finally merged to one G_{Tree} (Step 13).

5 Related Work

Goal modelling methods are applied in requirements engineering (RE) and process modelling. Goal modelling approaches have received much attention by researchers and practitioners [10] in RE since goals can be used as a mechanism for associating requirements to design and supporting reuse [17] and an effective way to identify requirements [15]. [10] developed an analysis framework for goal modelling approaches in order to systematically compare existing approaches and to highlight open issues in RE.

We will use this framework to compare our goal modelling approach with existing ones. The framework comprises the four views *usage*, *representation*, *subject* and *development*. The *usage view* concerns the objectives of using goal modelling in RE namely (1) understanding current organisational situation; (2) understanding the need for change; (3) providing the deliberation context within which the RE process occurs; (4) relating business goals to functional and non functional system components and (5) validating system specification against stakeholders' goals. The subject view abstracts three types of goal namely enterprise goals, process goals and evaluation goals. *Enterprise goals* focus on individual actors' goals or on business objectives or on a desired situation to reach in future, a *process goal* refers to anything that can act as a goal of the RE process, e*valuation goals* may refer to both the evaluation of the outcome of RE as well as the evaluation of the RE process.

The *representation view* concerns informal, semi-formal and formal notation of goals. Finally, the *development view* puts emphasis on the way goal models are generated and evolved. One can use the herein presented goal modelling approach to support understanding current organizational situation and to relate business goals to system components (*usage view*), since it analyzes given process models for goal elicitation. Further, it concerns enterprise goals (*subject view*) as goals extracted from process descriptions represent desired states or situations in future. Reference ontology and a hierarchical goal tree (*representation view*) formally describe extracted goals. Finally, goal models are automatically extracted from given EPC models (*development view*). Based on this classification and according to the overview of current goal-oriented research in RE given in [10], the RE-approaches i*/GRL [9], KAOS [6], GBRAM [1], Goal-based scenario coupling [19] and NFR framework [20] are similar to our approach but there are significant differences.

In the following, we explore the most import ones. The most commonly used notation for representing goal models is that of a goal decomposition tree (or graph) by using AND/OR decompositions for goals. Further, "*none of the approaches offer*

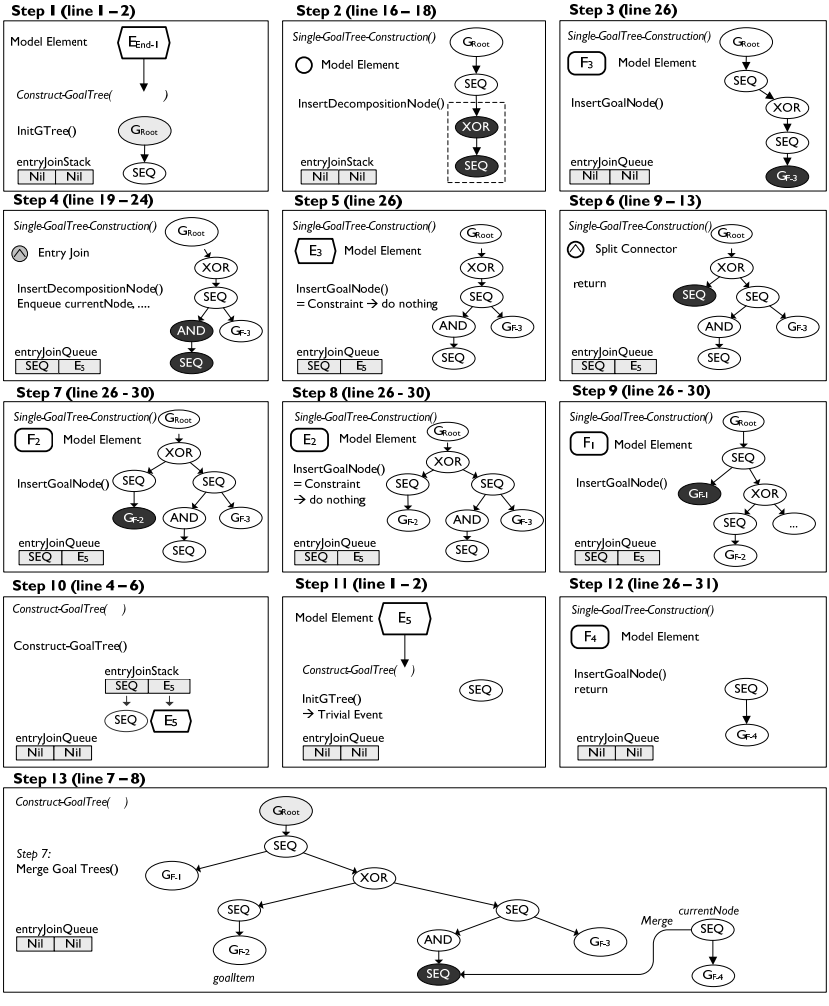


Fig. 10. Steps performed by the Algorithm for Example in Fig. 10

adequate methodological support to deal with the complexity of the goal analysis process. Goal analysis is assumed to be a well-structured process based on the analysis of existing documents or interviewing experts [10]".

Our approach deals with automated extraction of process goals from given EPC (goal analysis). It is an approach primarily defined for identifying common modelling practices in EPC models. This requirement gives rise to construct a goal model in terms of a hierarchical tree for each EPC model being analyzed for pattern identification. Common modelling practices are identified by matching similar goals of each tree. Similar goals and their decomposition relations form the base for specifying semantic relationships between process patterns. The specification of semantic relationships for process patterns additionally requires - in contrast to

existing approaches - to consider a sequence decomposition of goals for maintaining the satisfaction order of defined subgoals. This means that the satisfaction order of similar goals play a vital role for specifying sequence relationships between process patterns (e.g. pattern B *succeeds* to pattern A). Further, the meaning of an *"AND"* decomposition differs significantly with an *"AND"* decomposition used in existing approaches. Usually, an *"AND"* decomposition means to decompose goals in the sense of an *is part of* relation for a supergoal. In our work an *"AND"* decomposition permits to decompose goals into subgoals that can be satisfied concurrently.

Modelling goals from given process descriptions is realized by goal annotation approaches. Goal annotation of process models means to annotate process models and model fragments with goal ontology to specify the objectives of processes. Our work has common ideas with the ontology-based semantic annotation framework developed by Lin [12]. The framework uses three ontologies: it uses *general process ontology* for meta-model annotation, *domain ontology* for process model annotation, and *goal ontology* for process goal annotation. The framework also provides a goal annotation algorithm that performs a semi-automatic goal annotation. The logical connectors (or, and, xor) for a goal decomposition are not considered. The goal ontology is like a taxonomy of goal concepts that serve for a semantically-aligned goal representation. Our approach extends Lin's approach by adding an algorithm that extracts goals from process descriptions. The Lin's annotation algorithm simply matches activities assigned to goals of goal ontology with a semantically described activity of a process description.

6 Conclusions and Practical Experiences

In this paper, we presented an approach for the extraction and annotation of process goals for the EPC modelling language. According [12], the purposes of goal annotation are *"1) to enrich the semantics of the objectives of processes; 2) to provide a way to find the process knowledge based on business goals"*. Enriching process models with goal ontology facilitates to employ reasoning algorithms to identify common modelling practices in given process descriptions.

The research activities were carried out within the *BPI (Business Process Improvement)* project. The projects' objective was to identify *"common"* or *"best practice"* from given EPC models – designed by the ARIS Toolset - that describe product life cycle management (PLM) processes in engineering domains. We implemented the prototype *pModeler* that facilitates an automated identification of common modelling practices in EPC models. Practical experiences with our prototype have shown that an ontology-based description of process goals in combination with process patterns facilitate process modeller to locate context specific proven process solutions. Further, the goal-based approach has been proven as suitable rationale to identify process patterns that provide common pattern solutions in different engineering domains such as software or hardware development. The experiences also give rise to elaborate our goal ontology with additional concepts such as non-functional goals (Soft Goals) and additional relations between goals (*"influences", "supports", "conflict"*) as described by [7] and [14].

References

1. Anton, I.A.: Goal based requirements analysis. In: Proceedings of the 2nd International Conference on Requirements Engineering (ICRE 1996), pp. 136–144 (1996)
2. Baines, T., Adesola, S.: Developing and evaluating a methodology for business process management. Business Process Management Journal 11, 37–46 (2005)
3. Bögl, A., et al.: Semantic Annotation of EPC Models in Engineering Domains by Employing Semantic Patterns. In: Proceedings of the 10th International Conference on Enterprise Information Systems (ICEIS 2008), Barcelona, Spain, June 12-16 (2008)
4. Bögl, A., et al.: Knowledge Acquistion from EPC Models for Extraction of Process Patterns in Engineering Domains. In: Multikonferenz Wirtschaftsinformatik 2008 (MKWI 2008), München (2008)
5. Brash, D., Stirna, J.: Describing Best Business Practices: A Pattern-based Approach for Knowledge Sharing. In: Prasad, J. (ed.) Proceedings of the 1999 ACM SIGCPR Conference on Computer Personnel Research, SIGCPR 1999, New Orleans, Louisiana, United States, April 08-10, 1999, pp. 57–60. ACM, New York (1999)
6. Dardene, A., Lamsweerde, A., Fickas, S.: Goal-directed requirements acquisition. Science of Computer Programming 20(1-2), 3–50 (1993)
7. Giorgini, P., Mylopoulos, J., Nicchiraelli, E., Sebastiani, R.: Formal Reasoning Techniques for Goal Models. In: Spaccapietra, S., March, S.T., Kambayashi, Y. (eds.) ER 2002. LNCS, vol. 2503. Springer, Heidelberg (2002)
8. Gruhn, V., Laue, R.: How Style Checking Can Improve Business Process Models. In: INSTICC: 8th International Conference on Enterprise Information Systems (ICEIS 2006) (2006)
9. i*/GRL, An Agent-oriented modelling framework (2008), `http://www.cs.toronto.edu/km/istart/` (last visited: 25.07.2008)
10. Kavakli, E., Loucopulos, P.: Goal Modelling in Requirements Engineering: Analysis and Critique of Current Methods. In: Krogstie, J., Halpin, T., Siau, K. (eds.) Information Modeling Methods and Methodologies (Adv. topics of Database Research), pp. 102–124. IDEA Group (2004)
11. Keller, G., et al.: Semantische Prozessmodellierung auf der Grundlage, Ereignisgesteuerter Prozeßketten (EPK). In: Scheer, A.-W. (Hrsg.) Veröffentlichungen des Instituts für Wirtschaftsinformatik, Heft 89, Saarbrücken (August 20, 2007), `http://www.iwi.unisb.de/Download/iwihefte/heft89.pdf`
12. Lin, Y., Sølvberg, A.: Goal annotation of process models for semantic enrichment of process knowledge. In: Krogstie, J., Opdahl, A.L., Sindre, G. (eds.) CAiSE 2007 and WES 2007. LNCS, vol. 4495, pp. 355–369. Springer, Heidelberg (2007)
13. Mendes, R., et al.: Understanding A Strategy: a Goal Modeling Methodology. In: 7th International Conference on Object-Oriented Information Systems, Calgary, Canada, pp. 437–446 (2001)
14. Mylopoulos, J., Chung, L., Yu, E.: From object-oriented to goal-oriented requirements analysis. Communications of the ACM 42, 31–37 (1999)
15. Potts, C., Takahashi, K., Anton, A.I.: Inquiry-based requirements analysis. IEEE Software 11(2), 21–32 (1994)
16. Schuette, R., Rotthowe, T.: The guidelines of modeling - an approach to enhance the quality in information models. In: Ling, T.-W., Ram, S., Li Lee, M. (eds.) ER 1998. LNCS, vol. 1507, pp. 240–254. Springer, Heidelberg (1998)

17. Yu, E., Mylopoulos, J.: Why goal-oriented requirements engineering. In: Proc. of the 4th of International Workshop on Requirements Engineering: Foundations of Software Quality (1998), http://www.cs.toronto.edu/pub/eric/REFSQ98.html (last visited: 26.07.2008)
18. Lin, Y., Strasunskas, D., Hakkarainen, S.E., Krogstie, J., Solvberg, A.: Semantic Annotation Framework to Manage Semantic Heterogeneity of Process Models. In: Dubois, E., Pohl, K. (eds.) CAiSE 2006. LNCS, vol. 4001, pp. 433–446. Springer, Heidelberg (2006)
19. Rolland, C., Souveyet, C., Achour, C.B.: Guiding Goal Modeling Using Scenarios. IEEE Trans. Software Engineering 24(12), 1055–1071 (1998)
20. Mylopoulos, J., Chung, L., Nixon, B.: Representing and Using Nonfunctional Requirements: A Process-Oriented Approach. IEEE Transactions on Software Engineering SE-18(6), 483–497 (1992)

Part III

Information Systems Analysis and Specification

A Service Integration Platform for the Labor Market

Mariagrazia Fugini

Dipartimento di Elettronica e Informazione, Politecnico di Milano
Piazza da Vinci 32, Milano, Italy
fugini@elet.polimi.it

Abstract. Employment Services are an important topic in the agenda of local governments and in the EU due to their social implications, such as sustainability, workforce mobility, workers' re-qualification paths, training for fresh graduates and students. The SEEMP system presented in this paper overcomes the issue in different ways: starting bilateral communications with neighbor border similar offices, building a federation of the local employment services, and merging isolate trials.

Keywords: E-Employment, services for Public Administrations, Semantically Enabled Platforms, Data Mediation, Service Integration.

1 Introduction

Highly skilled human resources are more and more the key factor of economic growth and competitiveness in the information age and knowledge economy. But due to a still fragmented employment market, despite the enlargement of the EU, these resources are not effectively exchanged and deployed. The SEEMP (*Single European Market Place)*[1] Project brings a technical and business response to this problem, by enabling a federated market place of employment mediation agencies to interoperate through a peer-to-peer network based on employment data and mediation services.

The SEEMP enabled employment market place recognizes the essential needs of both *Public Employment Administrations* and *Private Employment Agencies* to:

- Provide citizen-centered services of employment promotion and regulation;
- Give market access to a large pool of human resources to maximize economic returns of staffing business.

It focuses on the job seekers and their anonymous (for privacy reasons) profiles as central trading resources to bring the key players of the job market onto the SEEMP market place for high visibility of job seekers and recruiters and effective matching of candidacy and vacancies. This paper focuses on the key technical and business aspects that SEEMPS introduces among the *public* and *private* employment mediators, facilitating the transparency and flow of information and creating ultimately significant social as well as economic impacts. The paper presents the main actions to be undertaken by an ES to get connected to the SEEMP platform. In particular, we give a

[1] SEEMP Project IST-4-027347-STP started Jan. 2005 and to be completed Dec. 2008. Project site: http://www.seemp.org/

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 445–455, 2009.
© Springer-Verlag Berlin Heidelberg 2009

set of "recipes" on how to configure and connect different systems, showing a pilot integration of an employment system, while currently SEEMP is built through the integration of EURES (Europe), TELMI (Italy), and EBP (Belgium). The SEEMP business vision, which we also illustrate, facilitates employment mediation to the benefit of Job Seekers, Employers and Job Advisors.

2 Background

The e-recruitment "industry" remains highly fragmented and is continually evolving. Speculators expect explosive growth in this sector due to three macroeconomics trends that are seen as fuelling the growth of this industry: 1) Shorter employment tenures, 2) Shrinking labor pools and 3) Need for hi-tech workers. A large number of online job portals have sprung up, dividing the online labor market into information islands and making it close to impossible for a job seeker to get an overview of all relevant open positions. In order to improve market transparency, several public bodies like the German Federal Employment Office [www.arbeitsagentur.de], the Swedish National Labour Market Administration (AMS) [www.ams.se], the Public Employment Agency of Wallonia, Belgium [http://www.leforem.be] and Regione Lombardia [www.borsalavorolombardia.net], Italy, have started projects to integrate open positions in a central database. In all projects, participating employers use terms from a controlled vocabulary to categorize their postings and send them to the central database using variations of the HR-XML[2] data format. Moreover, the employment mediation market is still characterized by a lack of transparency. For several reasons, there are no unique meeting points between candidates and employers. The need for interoperability at European Level among e-Government services has been perceived since 1999 [1] with the adoption of actions and measures for pan-European electronic interchange of data between administrations, businesses and citizens (IDABC) [2]. The main results of IDABC is the European Interoperability Framework (EIF) [3]. EIF follows the principle of subsidiarity[3] in addressing the interoperability problem at all levels: organizational, semantic and technical. One crucial aspect is to keep responsibility decentralized; each partner should be able to keep its own business process almost unchanged[4] and to provide externally point of exchange for its processes. EIF names these points "business interoperability interfaces" (BII). EIF does not prescribe any solution, but rather recommends the principles to be considered for any e-Government service to be set up at a pan-European level. SEEMP is proposed as an implementation of EIF in the domain of e-employment.

3 Collaboration Services

The SEEMP platform favors the meeting between Job Offers and Curricula (CVs) and improves the business activity of brokers in the market (public, private, off and on

[2] http://www.hr-xml.org/

[3] The principle of subsidiarity recommends no interference with the internal workings of administrations and EU Institutions.

[4] Quoting from IDABC: "it is unrealistic to believe that administrations from different Member States will be able to harmonize their business processes because of pan-European requirements".

line). The SEEMP solution is *collaboration*: a confederated network forming a market place where information is shared and traded on the recognition of existence of fragments. On this marketplace, services are shared and data exchanged with participants' consents. This mode of competition, named *coopetition* [4] of multiple players is becoming a pressing need among Regions and Agencies within and across Nations.

3.1 The Business and Technological Approach

SEEMP adopts a mix of methodologies that encompass both business and technological aspects. We briefly summarize the business aspects necessary to defined the designed technical solution.

The targeted customers of the SEEMP Technology Platform are the Employment Service Providers:

- *Public Employment Systems* (Governments, Public Job Agencies, Employment Offices). – These are named *PES.*
- *Private Employment Agencies.* – These are named *PrEA*.
- Education and training providers.

When there is no need for a distinction we use the term *Employment System (ES)* to denote both public and private systems.

Through their extended needs to access to the job market and information, the Employment Service *Consumer* will benefit of effective services to improve employment conditions. The Employment Service Consumer includes Job Seekers, Employers, Training Advisors and Job Counselors. The key needs on the political and economic agenda of the Public Employment Administration are the citizen-centered government services and efficient employment market, to simplify, how the Public Employment Administrations can effectively support citizens in finding a job, in spite of *geographical, organizational, linguistic and systemic barriers*. This includes enabling them to have visibility in the job market and to prepare their career change and advancement. The lifeline of the Private Employment Agency is not only having the employer but also a considerable repertoire of job seekers and profiles. Whether multi-national or regional, they face a serious obstacle of a fragmented job market.

SEEMP offers an interoperability platform for their collaboration. The *Employment Mediation Marketplace (EMM)* obtained through SEEMP is an "open space" providing an interoperability ICT infrastructure to support e-Employment mediators in participating for making business based on the interchange of services.

3.2 Consequences of Business Integration Needs

In order to analyze the technical vision, SEEMP partners followed well established methodologies in Software and Knowledge Engineering and Information Systems, with a specific attention to EIF.

The business aspects highlight the need for by-passing:

- *language heterogeneity*;
- *CVs and Job Offers structural heterogeneity*, i.e., the use of standards like HR-XML[5] is not wide spread and a multitude of local formats exist;

[5] http://www.hr-xml.org/

- *CVs and Job Offers content description heterogeneity*, i.e., European level occupation classifications like ISCO-88[6] exist, but they do not reflect differences and perspectives of political economic, cultural and legal environments; and
- *system heterogeneity* in terms of *service interface* and *behavior*: no standard exists for e-employment services.

3.3 Services in the Interoperability Framework

SEEMP relies on *Service Integration*, *Annotation*, and *Discovery*. Following the EIF, each *ES* locally must expose its Business Interfaces as a set of Web Services. SEEMP models a single consistent set of Web Services out of those exposed by the ESs.

SEEMP uses Mélusine [5] as a tool for modeling those abstract services and orchestrating the process of delegating the execution to the distributed independent service providers. Moreover, SEEMP relies on *Semantic Annotations* (both ontologies and mediators). Each ES has its own local ontology for describing at a semantic level the exposed Web Services and the structure/content of the exchanged messages. These ontologies are fairly similar, because a *common knowledge* about the *employment context* exists in SEEMP, as well as the needs for exchange. So, SEEMP, as a marketplace, models a single consistent ontology out of those exposed by the ESs. Such Reference Ontology becomes the actual standard for ESs that should provide the mediators with methods for translating from the local ontologies to the *Reference Ontology*, and vice versa. SEEMP adopts: WSMO [6] to semantically describe Web Services, ontologies and mediators; WSML [7] as concrete syntax for encoding those descriptions, and Methodology [8] as methodology for developing and maintaining those semantic descriptions.

A key point is *minimal shared commitment* in order to keep a ``win-win" situation among all ESs. This means that ESs are able to share while maintaining all the necessary and unnecessary disagreements. It may appear counter-intuitive, but the most suitable set of services and ontology is the one that enables actors to ``agree while disagreeing".

4 Technology Description

The SEEMP technical solution is composed of a SEEMP *reference part* (in the dotted-line box of Figure 1), which reflects the ``minimal shared commitment" both in terms of services and semantics, and by the *connectors*.

The *reference part* consists of *EMPAM* (Employment Market Place Abstract Machine) and of a set of *SEEMP services*. The EMPAM offers abstract services made concrete by *delegation mechanisms*: upon invocation of an abstract service, the EMPAM delegates its execution to the appropriate ES by invoking the correspondent *concrete services*. It acts as a multilateral solution (as request by EIF), where all the services connected to the EMPAM are made available to all ESs. Hence, they ensure a pan-European level of services with no interference with the business processes of each ES.

[6] http://www.warwick.ac.uk/ier/isco/isco88.html

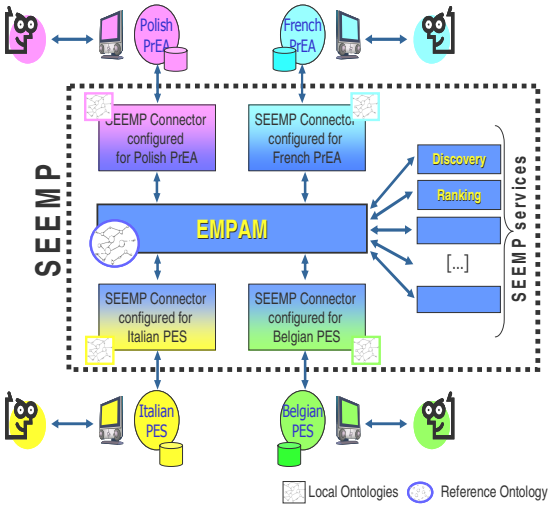


Fig. 1. A bird view of SEEMP solution

Two SEEMP *services* are depicted in Fig. 1. The *discovery* service is offered by Glue [9]. The EMPAM invokes the Glue Discovery Engine before delegating the execution to the concrete services exposed by the ESs. Glue analyzes the CV sent by the invoking PES and PrEA and selects those that most likely return relevant Job Offers. The *ranking* service is invoked by the EMPAM after all the concrete services have answered, and merges the results providing an homogeneous ranking of the returned Job Offers. It also deletes duplicated Job Offers.

The SEEMP *connectors* enable communications between the EMPAM and a given PES and PrEA. A SEEMP connector is available for each connected PESs and PrEAs and has two main tasks:

- *Lifting and Lowering*: when communicating with the ES any outgoing (or incoming) data that are exchanged by the means of Web Services must be lifted from XML to WSML in terms of the local ontologies of the PES and PrEA (or lowered back to the XML level from WSML).
- *Solving Heterogeneity*: each ES has its own local ontology that represents its view on the employment domain. The SEEMP connector is responsible for resolving these heterogeneity issues by converting all the ontologized content (the content lifted from the XML received from the ES) into content in terms of the reference ontology shared by all partners and vice versa.

In SEEMP these tasks are achieved through an extension to R_2O language [10], which enables to describe mappings between XML schemas and ontologies, an extension to its related processor ODEMapster [11] and the use of WSMX data mediation [9].

As an example, consider the following problem:

> *"Job seekers (companies) put their CVs (Job Offers) on a local ES and ask to match them with the Job Offers (CVs) other users put in different ESs through SEEMP".*

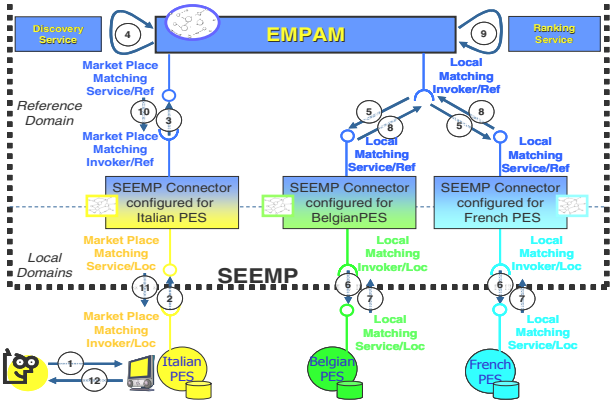


Fig. 2. SEEMP technical solution to the matching problem

Although fairly simple, this example shows that, in order to reach the potential EU-wide audience, a local matching service designed for national/regional requirements only is not sufficient (i.e., central database, single professional taxonomy, single user language, etc.), while SEEMP is able to send the request, which an end-user submits to the local ES to all the other ESs in the marketplace. In order to avoid asking ``all'' actors, it has to select those that most likely will be able to provide an answer and send the request only to them. Moreover, the answers should be merged and ranked homogeneously by SEEMP before they are sent back. In Figure 2 this problem is addressed by combining the EMPAM and the SEEMP connectors [9]. The solution enables a meaningful service-based communication among ESs that follows the steps depicted in the figure.

4.1 Involving a New Node

One problem that the SEEMP structure is expected to solve is to enable a smooth integration of new nodes. There can be several cases, depending on the technical structure of the new node [11].

1) *The new ES is endowed with Web Services.* In this case, two tasks must be undertaken by the engineer wishing to build a Connector. Firstly, the engineer must identify the mapping between the ES services and those expected by the EMP and link these together via some code. In other words when the EMP invokes the *matchCV* interface of the connector, this invocation must result in the invocation of one or more of the interfaces of the ES. Of course the reverse mapping must also be made such that invocations from the ES are translated into invocations to the EMP. Thus the Connector must expose services upwards towards the EMP and downwards towards the ES. While the services exposed to the EMP are standardized, the choice of the services to expose to the ES is at the discretion of the Connector architecture.

2) *The ES has no Web Services*. The first approach that could be taken by a given ES is to expose their own local Web Service interfaces and then to employ the Connector architectures. However, there are other approaches that may be more advantageous for the ES in terms of performance and scalability, depending on the internal

architecture of the ES. In this case, the potential architectures belong to two categories, namely a *light* solution or *recipe*, as we call it in SEEMP, and a *heavy* recipe, denoting the size of the change to the internal architecture of the ES by employing these architectures.

Using a *Light Architecture*, we have two possibilities:

A) *XML-based Light Architecture*
Within the ES, a number of software components will exist for functionality such as matching of candidacies to vacancies, matching vacancies to candidacies etc. In this light architecture, we reuse these software components in order to perform the necessary functionality required when the EMP invokes the ES through the connector. The architecture of the Connector must identify which Web Service interfaces on the Connector map to which software components within the ES, in a similar manner to the previous case of mapping Connector Web Services to ES Web Services. Once this mapping is understood by the architect we need to translate the data returned from the software components into instances of the SEEMP Reference Ontology in order to fulfil the minimal commitment of semantics.

B) Ontology based Light Architecture
Rather than blindly lifting to XML and then lifting to the ontological level it can be more advantageous to lift the results of the software components directly to instances of a local ontology. This approach has the advantage that additional software components could be written within the ES that can also benefit from the semantics within the local ontology for building more advanced algorithms for candidacies and vacancies matching etc.

Using a *Heavy Architecture* is another possibility. While in the light architecture internal changes to the ES are relatively minimal, it is possible to go further, if the ES wishes to make their entire system more *semantic based*. The heavier approach involves the lifting of data within the ES to instances of a local ontology or the Reference Ontology at a very early phase. This lifting also has the knock on effect of requiring a modification of existing software components to handle these new ontological instances instead of standard software artefacts. The major advantage is that existing algorithms for matching candidacies and vacancies can be enhanced with the richer semantics offered by the ontologies.

3) *Employment Services with no software infrastructure.* Such employment system needs to set up an infrastructure for candidacies and vacancies storage and for making these data available to the EMP network. Such ESs are in a unique position to take advantage of the Reference Ontology and adopt it as their view on the world of employment.

In Table 1 below we give an overview of the architectures provided through this section.

4.2 An Example

We show the Integration of SEEMP Services into the EURES system (http://www.europa.eu.int/eures/). The customizations and enhancements that have been performed on EURES to achieve the final integration are detailed below:

1. *Data*: The set of data that has been used with the SEEMP-EURES pilot instance consists of real (though outdated) information extracted by the EURES instance in production. Any sensitive information is hidden not only at the GUI level but also at the database level.
2. *Non-visible Components*: A new set of Web Services has been developed which interact with the EURES Connector to enhance EURES with the SEEMP services. The newly supported functionalities are: a) Search for candidacies through the SEEMP network; b) Match a candidacy against vacancies; c) Search for vacancies through the SEEMP network; d) Match a vacancy against candidacies.

Table 1. Overview of Connector architectures

Category	Recipe	Advantages	Disadvantages
Web Service	Direct XML Mapping	Simple Architecture decoupled from ES internal architecture	Not feasible for complex heterogeneity problems
	Two Step Syntactic Transformation	Simple architecture with current technologies decoupled from ES internal architecture	Mappings are very complex and hard to maintain, Reference XML format must be maintained
	Two Step Semantic Transformation	Much simpler mappings due to syntactic and semantic steps, decoupled from ES internal architecture	Local ontology must be maintained
No Web Services	Light Architecture	Little change to existing infrastructure	Ontology generator needs to be hand coded Architecture is tightly coupled to internal ES architecture
	Heavy Architecture	Existing database to ontology tools can be used	Large change to internal infrastructure Architecture is tightly coupled to internal ES architecture
No Infrastructure	Adopting Reference Ontology	Existing database to ontology tools can be used	Architecture is tightly coupled to internal ES architecture Changes to Reference Ontology must be reflected in internal ES architecture

3. *Visible components (GUI)*: the web application's graphical interface has been extended to support entering new search criteria (wherever applicable) to invoke SEEMP services and new screens that display the returned results. Integrated EURES in View of Users.

In what follows, some screenshots help visualize the changes in the EURES system.

a. Search for candidacies: A EURES advisor can use the new enhanced GUI to search for candidacies in SEEMP (Figure 3).

b. After the search, the user can browse the results as in a EURES search. The new functionality for Matching the candidacy with vacancies is available via a new button (Figure 4).

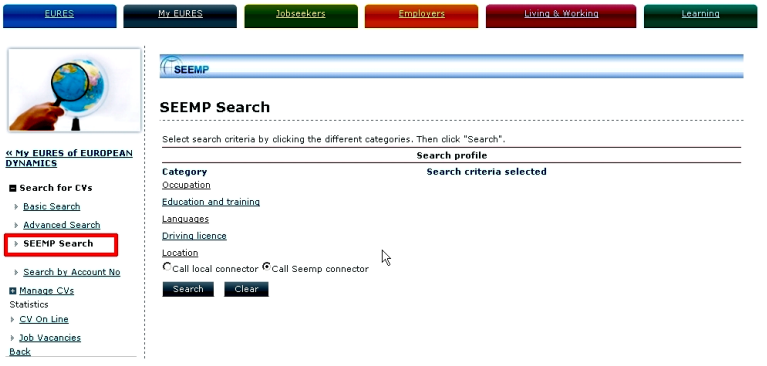


Fig. 3. SEEMP search for EURES advisors

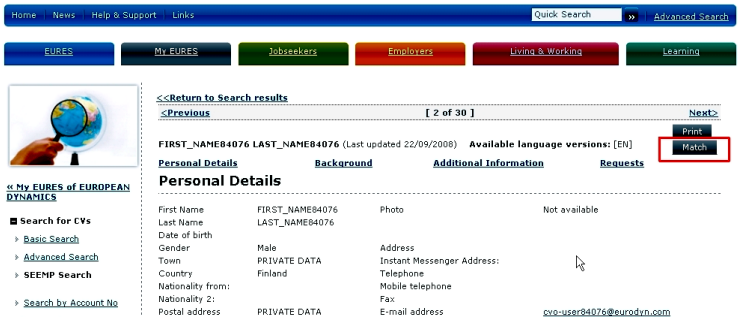


Fig. 4. SEEMP results of the Match functionality

a. Match candidacy to vacancies: After the user clicks on the Match button the Match operation is invoked. The operation returns all the vacancies matching the selected candidacy (Figure 5).

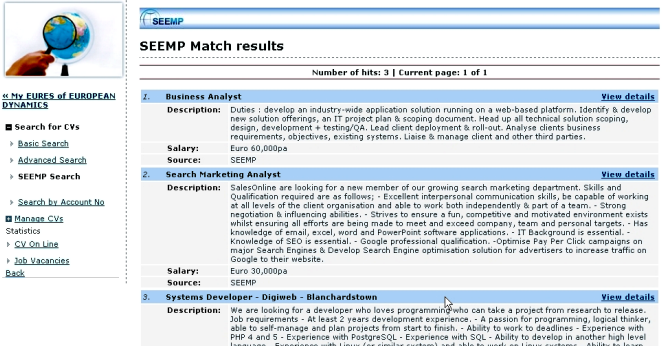


Fig. 5. SEEMP results when matching a candidacy

b. Search vacancies: A public user can use the customized EURES instance to search for vacancies (Figure 6). All search criteria can be filled-in in one page.



Fig. 6. Search for vacancies using SEEMP

After the search invocation, the vacancies that match user's criteria are displayed. When the user clicks the details of a vacancy, a new option is enabled for matching this vacancy with candidacies.

c. Match vacancy to candidacies: The user refines the criteria for the occupation to match, using a drop-down. The Match functionality is automatically invoked and matching candidacies are returned (Figure 7).

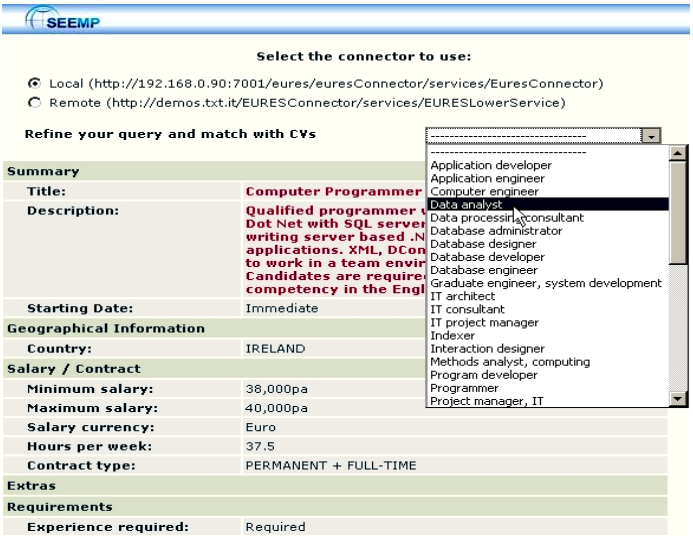


Fig. 7. Matching a vacancy with candidacies using SEEMP

5 Concluding Remarks

This paper has presented the main issues of SEEMP, a market place of mediators where public and private actors collaborate and complement each other. Final beneficiaries of their "win-win" situation are job seekers and employers with significant social and economic impact. Services and Semantics are the key concepts for abstracting from the hundreds of heterogeneous systems already in place that are evolving separately. They provide a straight forward way to implement the subsidiarity principle of EIF.

Acknowledgements. We acknowledge the partners in SEEMP Project, in particular, European Dynamics and CEFRIEL-POLIMI.

References

1. 1720/1999/EC: Decision of the European Parliament and of the Council of 12 (July 1999)
2. 2004/387/EC: Decision of the European Parliament and of the Council on Interoperable Delivery of pan-European Services to Public Administrations (2004)
3. European Communities: European Interoperability Framework for Pan-European eGovernment Services. Technical Report, Office for Official Publications of the European Communities (2004)
4. Cesarini, M., Celino, I., Cerizza, D., Della Valle, E., De Paoli, F., Estublier, J., Fugini, M.G., Guarrera, P., Kerrigan, M., Mezzanzanica, M., Gutowsky, Z., Ramìrez, J., Villazón, B., Zhao, G.: SEEMP: A Marketplace for the Labor Market. In: Proc. E-Challenges Conference, The Hague (2007)
5. Estublier, J., Vega, G.: Reuse and variability in large software applications. In: ESEC/SIGSOFT FSE (2005)
6. Fensel, D., Lausen, H., Polleres, A., de Bruijn, J., Stollberg, M., Roman, D., Domingue, J.: Enabling Semantic Web Services – The Web Service Modeling Ontology. Springer, Heidelberg (2006)
7. de Bruijn, J., Lausen, H., Polleres, A., Fensel, D.: The web service modeling language: An overview. In: Sure, Y., Domingue, J. (eds.) ESWC 2006. LNCS, vol. 4011, pp. 590–604. Springer, Heidelberg (2006)
8. Gomez-Perez, A., Fernandez-Lopez, M., Corcho, O.: Ontological Engineering. Springer, Heidelberg (2003)
9. Della Valle, E., Cerizza, D.: The mediators centric approach to automatic web service discovery of glue. In: MEDIATE 2005. CEUR Workshop Proceedings, vol. 168 (2005) CEUR-WS.org
10. Barrasa, J., Corcho, O., Gomez-Perez, A.: R_2O, an extensible and semantically based database-to-ontology mapping language. In: Bussler, C.J., Tannen, V., Fundulaki, I. (eds.) SWDB 2004. LNCS, vol. 3372. Springer, Heidelberg (2005)
11. Della Valle, E., Cerizza, D., Celino, I., Estublier, J., Vega, G., Kerrigan, M., Ramírez, J., Villazon, B., Guarrera, P., Zhao, G., Monteleone, G.: SEEMP: Meaningful service-based collaboration among labour market actors. In: Abramowicz, W. (ed.) BIS 2007. LNCS, vol. 4439, pp. 147–162. Springer, Heidelberg (2007)

Developing Business Process Monitoring Probes to Enhance Organization Control

Fabio Mulazzani, Barbara Russo, and Giancarlo Succi

Free University of Bolzano, Faculty of Computer Science
Via della Mostra 4, 39100 Bolzano, Italy
{fabio.mulazzani,barbara.russo,giancarlo.succi}@unibz.it

Abstract. This work present business process monitoring agents we developed called Probes. Probes enable to control the process performance aligning it to the company's strategic goals. Probes offer a real time monitoring of the strategic goals achievement, also increasing the understanding of the company activities. In this paper Probes are applied to a practical case of a bus company. Probes were developed and deployed into the company ERP system and determined a significant change in the strategy of the company and a corresponding enhancement of the performances of a critical business process.

Keywords: ERP Systems, Monitoring Agents, Probes, Business Strategy, Business Processes.

1 Introduction

Chief information officers and IT executives consider Strategic Alignment (SA) as a top priority in today's competitive market [11].

In literature various definitions have been given to the concept of SA. Reich and Benbasth state that SA is "the degree to which the IT mission, objectives and plans support and are supported by business mission, objectives and plans" [9], while Henderson and Venkatraman in [6] provide a comprehensive framework to approach the alignment concept. In their framework, two distinct relationships are described: "strategic fit" and "functional integration". Strategic fit is the external relationship concerned with the harmonization of business strategy choices (e.g. business scope, partnerships, alliances) and strategic choices concerning IS/IT deployment. Functional integration is the corresponding internal relationship concerned with organizational infrastructure and processes and IS/IT infrastructure and processes.

As stated in [4] a crucial activity that helps managers to have an organization strategically aligned is the monitoring of the performances of their Business Process (BP). In fact, BP are designed and developed to satisfy a precise Business Strategy (BS), hence, the performance of a BP must conform or respect the related BS. It becomes clear that IT could play a fundamental role in the monitoring activity because it would provide managers with automated and ubiquitous instruments that allow controlling the performance of the BP and its alignment with the BS expectations anytime they want.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 456–466, 2009.
© Springer-Verlag Berlin Heidelberg 2009

There are different approaches that can be used to identify what to measure of a BP, yet, unfortunately, they do not define how to relate the performance of the BP to its BS [1].

In this paper we address the problem of relating process performances to a BS presenting BP monitoring agents called Probes. Probes are monitoring agents deployed into the IT infrastructure that operate over a process, or part of it (e.g. a group of tasks), to control if it respects the organization strategic goals, objectives or constraints. We determined how to develop the Probes combining two engineering techniques, the Goal Question Metric (GQM) approach [2] and the Business Motivation Model (BMM) [8]. From an IT perspective, it is essential defining a method to identify these monitoring agents as it becomes easier to deploy them into the organization IT infrastructure, such as an Enterprise Resource Planning System (ERP) [5, 10]. Probes are included into the framework SAF [4] (Strategic Alignment Framework) that we developed in order to help IT Analyst to better understand the context of a company, thus helping them to: (i) align/make coherent organizations BP to organizations BS; (ii) align organizations IT infrastructure to organizations BP; (iii) create a strategically aligned IT infrastructure for those organizations newly founded.

The paper exemplify (i) how Probes are developed and (ii) how Probes act into a real case study of a bus transportation company.

The rest of the paper is structured as follows. In section 2 we introduce the concept of business process monitoring and we describe the state of the art in this field summarizing the most common techniques used by IT practitioners. In section 3 we describe the case study we considered. In section 4 we illustrate the approach we developed to create the Probes, and offer a practical example of Probes development. Finally we make our conclusions and identify the future works.

2 Theoretical Background

In the field of business process monitoring there are several techniques that aim to define the correct metrics that enables a high understanding of monitored process.

In this section, we review the two top techniques used to monitor a process, namely the Balanced Scorecards and the Goal Question Metric approach. We also describe the Business Motivation Model, a formal technique developed to describe the strategic intents of a company, we used to relate the monitored performance of a process to its strategy.

2.1 The Balanced Scorecards

Balanced Scorecards (BSc) were defined by their developers, Robert S. Kaplan and David P. Norton, as a multidimensional framework for describing, implementing and managing strategy at all level of enterprise by linking objectives, initiatives and measures to organization's strategy [7]. The BSc provides a framework for studying a causal link analysis based on internal performance measurement through a set of goals, drivers and indicators grouped into four different perspectives: (i) Financial; (ii) Customer; (iii) Internal processes; (iv) Learning and growth.

The weakness of BSc is that its framework does not provide a constructive way to implement the strategy into the operational level. For example even though they use Key Performance Indicators (KPI) they do not discuss a method to derive and identify them.

2.2 The GQM and Its Evolution to the GQM$^+$Strategy

The Goal Question Metric approach [2] provides a top-down paradigm for an organization or a project to define goals, refine those goals down to specifications of data to be collected, and then analyze and interpret the resulting data with respect to the original goals. GQM goals are defined in terms of purpose, focus, object of study, viewpoint, and context. Such a goals are then refined into specific questions that must be answered in order to evaluate the achievement of the goal. The questions are then operationalized into specific quantitative measures. The GQM formalizes the deduction process that derives the appropriate measures to answer a given business goal. Traditionally, the GQM has been used in the measurement of the software development processes.

Although it is a powerful instrument in measurement theory, the GQM lacks of an explicit support for integrating its measurement model with other organizational elements such as higher-level business goals, strategies, and assumptions [3]. That is why Basili et al. in [3] propose and describe a method that adds several extensions on top of the GQM model, thus developing the GQM$^+$Strategy.

The GQM$^+$Strategy method makes the business goals, strategies, and corresponding software goals explicit. The GQM$^+$Strategy method also makes the relationships between organization activities and measurement goals explicit. Sequence of activity necessary for accomplishing the goals are defined by the organization and embedded into scenarios in order to achieve some related goals. Links are established between each goal and the business-level strategy it supports. Attached to goals, strategies, and scenario at each level of the model is information about relationships between goals, relevant context factors, and assumptions.

One of the weak point of the GQM$^+$Strategy is that it is too software process specific and a refinement in its strategic overlayer is needed to use it in a more wide context.

2.3 The Business Motivation Model

The Business Motivation Model [8], according to its developers, the Business Rules Group, is a meta-model of the concepts essential for business governance.

The BMM provides:

1. A vocabulary for governance including such concepts as “Influence”, “Assessment”, “Business policy”, “Strategy”, “Tactic”, “Goal”, and fact type that relate them, such as “Business policy governs Course of Action”.
2. Implicit support for an end-to-end process that runs:
   a. From recognition that an influencer (regulation, competition, environment, etc.) has an impact on the business;
   b. To implementing the reaction to that impact in business processes, business rules and organization responsibilities.
3. The basis for logical design of a repository for storage of BMMs for individual businesses.

There are two major components of the BMM. The first is the Ends and Means of business plans. Among the Ends are things the enterprise wishes to achieve - for example, Goals and Objectives. Among the Means are things the enterprise will employ to achieve those Ends - for example, Strategies, Tactics, Business Policies, and Business Rules. The second is the Influencer that shape the elements of the business plans, and the Assessments made about the impacts of such Influencers on Ends and Means (i.e. Strengths, Weaknesses, Opportunities, and Threats).

In its specifications BMM define KPI as a metric particularly important, and suggest the use of KPI to monitor the performance of processes. Unfortunately, BMM does not suggest how to link the strategic level it defines to the operational level of the processes, furthermore BMM does not provide a way to develop such a KPI

3 The Case Study

Dolomitesbus is (a pseudonym for) a public transportation company operating in a province of Northern Italy. Dolomitesbus has a bus .fleet of 290 units, and serves 60 routes connection every day. Each route is covered by two buses, one that works from 6 a.m. to 2 p.m., the other from 2 p.m. to 10 p.m.

The mission of the company is to offer to the customer a high quality transportation service over the province territory. The fleet is subjected to an extreme mechanical wear due to the mountain routes that serves, hence the main issue for the company is to concentrate a lot of resources on the maintenance process in order to have efficient buses. The mechanical workshop, that is part of Dolomitesbus, is in charge of any type of maintenance operation (i.e. planned maintenance or damage repair), and it also has to guarantee at least two fully operative buses every day for each route in accordance to the main quality constraint given by the Provincial Council - that is the supervisor for the company activities. The maintenance tactic imposed by the managers is the following: “Make a planned maintenance to every bus at most once per year - hence respecting the minimum requirement of the law - every other maintenance has to be made only when a problem occurs and needs to be repaired”.

Furthermore the company has defined a feedback mechanism where the customers can report their complaints. Till the end of 2007 the company has never inspected and properly used the resulting complaints.

For the last decade, the buses capacity reaches its critical load in the period of June/August. As such, the high number of passengers that are served in this period affect the company’s activities in two ways: (i) Increase the mechanical waste of the buses - due to the increase of the weight loaded; (ii) Increase the attention over the quality of the service offered - due to the increase of the customers transported.

Dolomitesbus has an efficient IT department that has always supported all the company’s software needs also adopting and developing Open Source Software. At the time the case study started, the company had its own in house ERP system that covered the activities of its departments but the mainte-nance process in the workshop. The operations of maintenance and repair done over a bus where only recorded manually with a paper schedule (containing fields such: bus number, type of operation, operation starting date, operation end date, details, etc.), not always properly filled, and then stored into an archive.

As the first step of this phase we interviewed the main responsibles of Dolomitesbus in order to better understand the various aspects of the company context in the maintenance process. For this we developed the GQM to characterize the maintenance business process.

1. Measurement Goal: Analyze the bus maintenance process in the context of the company workshop to characterize it in terms of number of buses under maintenance.
1.1. Question: How many maintenance operations are performed periodically by type of operation?
1.1.1. Metric: Number of maintenance operations per day per type.
1.1.2. Metric: Yearly number of maintenance operations per type.
2. Measurement Goal: Analyze the bus maintenance process in the context of the company workshop to characterize it in terms of time spent to operate over a bus.
2.1. Question: How much does each type of operation last?
2.1.1. Metric: Time duration per type.

The information needed to answer to the questions of the GQM were retrieved from the paper schedule. As mentioned, the schedule was designed with the following fields to be filled:(i) Bus number; (ii) Current Km; (iii) Operation Starting/Ending Date (hh.dd.mm.yyy); (iv) Total hours spent for the operation; (v) Type of operation - checkbox chosen between planned maintenance and damage repair; (vi) Detailed description of the operations; (vii) Mechanic responsible for the operation.

The paper schedule was used with the purpose of having an historical record of the operations done over the buses.

During the conversion to the electronic format of the paper schedules we found out that some fields were not properly filled or not even considered at all. The fields left blank were: Current Km; Hours of starting and ending date; Total hours spent for the operation. The fields not properly filled were: Detailed description of the operations - when filled it included a list of material used for the operation instead of a detailed description of the problem encountered; Mechanic responsible for the operation - it included an unreadable signature but not the cursive name and surname. The fact that the workshop mechanics have never filled the schedules in a proper way prove the fact that these have received a scarce importance or consideration. Namely, the schedules have never been used for any managerial analysis to establish the performance of the maintenance process. Furthermore, the bad compilation of the schedules has determined the permanent loss of important information that would have been useful for our analysis.

The results obtained by the application of the GQM are summarized in Table 1.

We then reported to the company managers our suggestions, as reported in detail in the following section.

3.1 Data Analysis

The data collected for Metrics 1.1.1 shows that the median number of buses daily operated is 40 for those under a planned maintenance and 100 for those under a maintenance for damages. In figure 1 is represented a line chart of (i) the daily number of buses having planned maintenance, (ii) the daily number of buses having a maintenance for damages, and (iii) the sum of the previous two. In figure 1 we also

Table 1. Metrics of the GQM collected

Description (PM: Planned Maintenance; MfD: Maintenance for Damages)	**2006**	**2007**	**'06/'07**
1.1 Question: How many maintenance operations are done each day divided by type of operations?			
Mean number of buses daily operated for PM	40	33	41
Median number of buses daily operated for PM	45	48	40
Mean number of buses daily operated for a MfD	90	91	101
Median number of buses daily operated for a MfD	115	111	100
1.2 Question: How many maintenance operations are done each year divided by type of operation?			
Total number of buses yearly operated for a PM	370	330	350
Total number of buses yearly operated for MfD	2360	2200	2280
2.1 Question: How much does each type of operation last?			
Mean duration in days for operations of type PM	39	69	52
Median duration in days for operations of type PM	7	16	9
Mean duration in days for operations of type MfD	13	20	16
Median duration in days for operations of type MfD	1	1	1

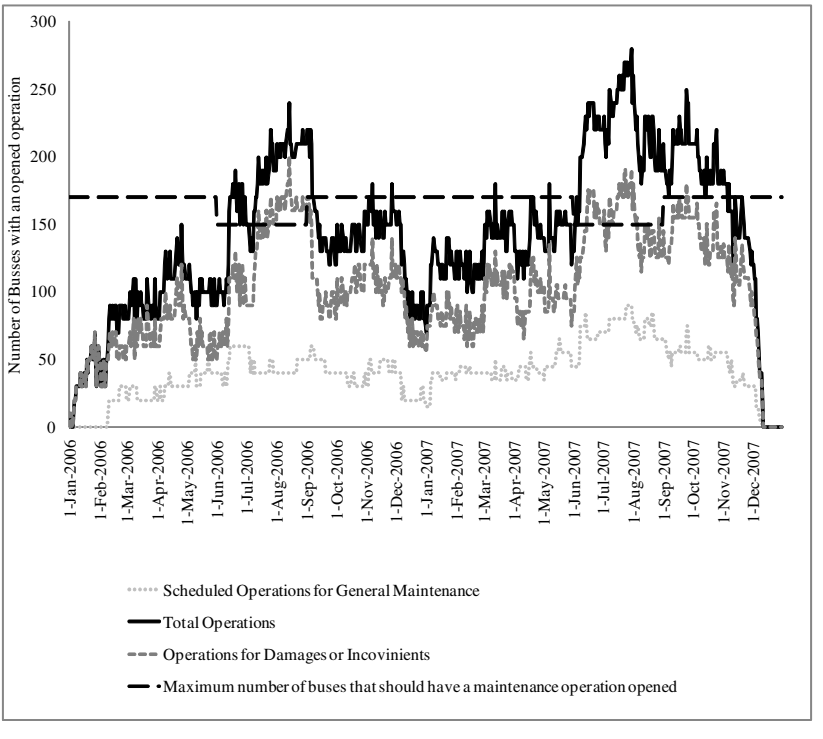


Fig. 1. Number of Maintenance Operations done in 2006 and 2007

show the threshold of the maximum number of buses that should be under maintenance every day without affecting the company mission. Dolomitesbus should have 120 buses (2 buses for each of the 60 routes) fully maintained every day in order to offer to its customers a high quality service, hence the maximum number of buses under maintenance every day is 170 (given by the 290 buses of the fleet minus the 120 fully maintained each day).

From June until the end of August the threshold is of 150 buses, since the high number of customers in that period requires an increase to 140 buses fully maintained each day. If we consider these two levels for the thresholds as a quality constraint over which the quality of the service is not guaranteed, we counted that in 2006 the limit was exceeded for 63 times (55 during summer time and 8 during the rest of the year), while in 2007 the limit was exceeded for 151 times (85 during summer time and 66 during the rest of the year). That means that for 214 days during the biennium 2006-2007 the routes were served with buses not completely maintained and that could have negatively affected the customers' quality perception. As confirmed by the managers feedbacks, during that particular days some of the 60 routes where covered by buses with a maintenance operation opened. In those cases the workshop manager selected the buses with the less significant inefficiency in order not to affect the road and passengers security.

The data collected for Metrics 1.1.2 shows that in the years 2006 and 2007 the mean of total planned maintenance operations is 350, and the mean of total operation for maintenance for damages is 2280. Considering these numbers over the bus fleet we can notice that every year each bus is subjected to a mean of 1,2 operations for planned maintenance and 7,6 operations for maintenance for damages.

The data collected for Metrics 2.1.1 shows that the mean duration for the planned maintenance operation is 52 days, while the mean duration for the operation of type maintenance for damages is 16 day. That means that every bus spend a mean of 69 days per year with an unsolved inefficiency - calculated multiplying the mean duration of the operation per the mean number of operation that a bus is subjected each year. Figure 1 clearly shows a different trend on the number of maintenance during 2006 and 2007. The managers of Dolomitesbus expected a difference between the two years, and they knew the causes. In fact, during an inspection at the company workshop at the end of 2006 they found out that the workshop managers did not spend much attention in filling the maintenance paper schedule. After the inspection, the top management imposed to the workshop manager more accuracy in filling the schedule and wrote down and defined a first attempt of business strategy related to the maintenance process. In section 6.1 in figure 3 we modeled the business strategy for the year 2007.

4 Developing the Probes for Our Case Study

In this section, we show the approach we defined to develop the monitoring Probes applied to the case study explained in section 3.

After the managers defined a new business strategy for the maintenance process and after Dolomitesbus IT Department implemented the new functionalities into the ERP system we defined the type of Probes useful for the managerial control.

To develop the Probe we first identified the business strategy that governs the maintenance process by modeling it with a standard notation called Business Motivation Model as described in Fig. 2. On the top of the figure is represented the mission of the company that can be achieved by means of some strategies. In the figure we included the specific strategy that refers to the maintenance process. On the bottom of the figure we then described the tactics, and the relative constraints, that the company needs to respect to fulfill the above strategy. On the left side we show the tactics used during the year 2007, while on the right side we show the tactic developed after our intervention at the end of 2007.

For the year 2008 the managers decided to adopt three new tactics with the relative constraints related to the maintenance process. Here follows the list of the new tactics and constraints as reported in figure 2:

- Tactic one (T1) imposes that of the 290 buses of the fleet only 240 are maintained by Dolomitesbus workshop, the remaining 50 needs to be maintained by a different workshop. The constraint for this tactic tells that the external workshop must guarantee at least 40 buses fully maintained every day, while the internal workshop must guarantee at least 90 buses fully maintained every day.
- Tactic two (T2) imposes that in case one of the threshold given in the constraint for the tactic one is not respected then two or three routes must be aggregated in couples or triples. The aggregated routes must be served by a bus that works over its time limit.
- Tactic three (T3) imposes that the manager must revise tactic one in case the company receives too many customers complaints. The constraint for this tactic is given by a threshold risk of 5%.

We have adopted as additional overlayer of the GQM$^+$Strategy the tactics that the managers defined for 2008 and we have then applied the GQM method to determine the metrics to be monitored. The logic of the threshold for the reporting function of the Probes has been derived from the business constraints that governs the relating tactics. Here it follows the structure of the GQM that determines the metrics to be measured by the Probes and the related threshold.

-Probe 1 - Measurement Goal (MG) for T1: Analyze the maintenance process from the point of view of the managers in order to understand the level of maintained buses in the context of Dolomitesbus Workshop. Question 1.1: How many buses are under maintenance at Dolomitesbus workshop every day divided by typology of maintenance? Metric 1.1.1:(Absolute) Number of Buses by day and typology. Threshold for this metric: According to CfT1 the max limit is 150.

-Probe 2 - MG for T1: Analyze the maintenance process from the point of view of the managers in order to understand the level of maintained buses in the context of External Workshop. Question 2.1: How many buses are under maintenance at the external workshop every day divided by typology of maintenance? Metric 2.1.1: (Absolute) Number of Buses by day and typology. Threshold for this metric: According to CfT1 the max limit is 10.

-Probe 3 - MG 1 for T2: Analyze the maintenance process from the point of view of the managers in order to understand the level of aggregation of bus routes caused by the unavailability of maintained buses (if there are not enough buses available the

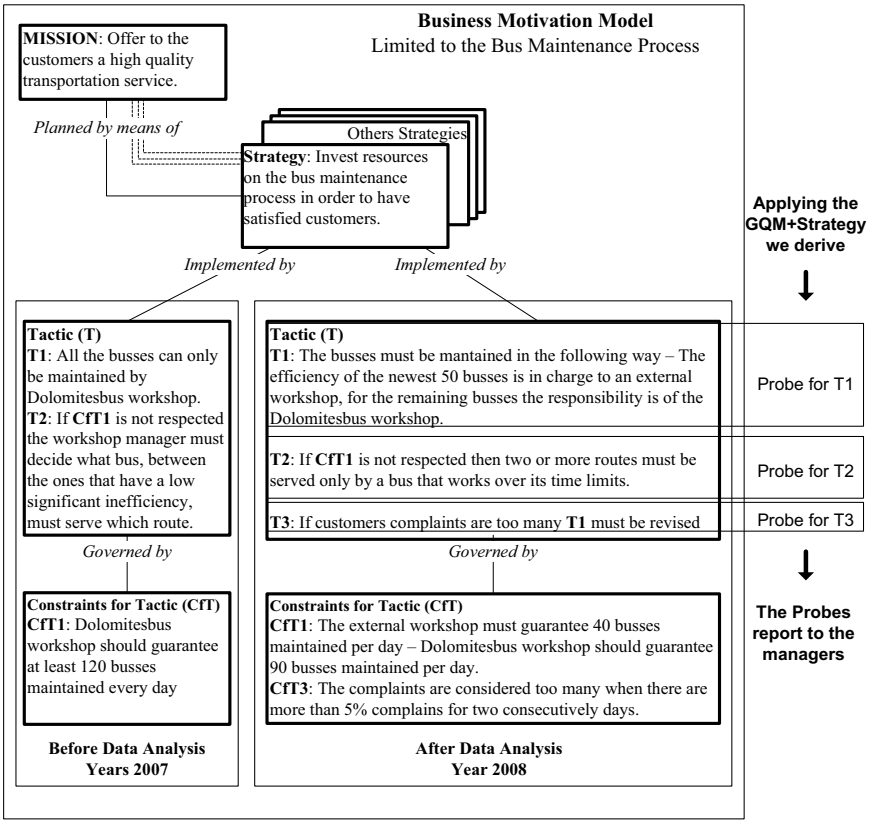


Fig. 2. Defining Probes

routes can be aggregated in couples or triples and served by a bus that works over it time limit). Question 3.1: How many bus routes are aggregated each day distributed by couples or triples? Metric 3.1.1: (Absolute) Number of bus routes. Threshold for this metric: The max limit can be defined after a simulation.

-Probe 4 - MG 1 for T3: Analyze the maintenance process from the point of view of the managers in order to understand the level of customers complaints. Question 4.1: How many complaints there are every day? Metric 4.1.1: (Absolute) Number of Complaints. Threshold for this metric: Below a given threshold risk (i.e. 5%).

Probes were developed using the model in figure 2. It is now under development a functionality that will send a warning mail to the managers if the indicators reach a below-threshold value.

5 The New Business and IT Set Up

In this phase we measured the performance of the maintenance process during the first three quarters of 2008.

Table 2. Metrics collected by the probes in the first three quarters of 2008

Probe	Metric	PM '08	MfD '08	PM+MfD '08	'06/'07	Δ
Probe 1	Metric 1.1.1	20	50	70	N.A.	N.A.
Probe 2	Metric 1.1.2	6	4	10	N.A.	N.A.
P. 1 & 2	Metric 1.1.1+1.1.2	26	54	80	140	-43%

Table 2 shows the metrics values collected by the probes developed in section 6.1. The values collected in 2008 were compared, if possible, to the average value of 2006/2007.

Probe 1 has measured that every day inside Dolomitesbus workshop there is an median of 70 buses that are subjected to a maintenance process.

Probe 2 has measured that every day inside the external workshop there is a median of 10 buses that are subjected to a maintenance process.

Summing the metric collected by the probes 1 and 2 we observe that the adoption of new business strategies defined by the company managers have determined a drastically decrease of 43% of the buses that are under any type of maintenance every day. That means that the choice of externalizing part of the maintenance process has determined a decrease not linear of buses present in the workshop every day for a maintenance. Furthermore, during 2008 Dolomitesbus have never exceeded the maximum threshold of buses that daily could be under maintenance.

Probe 3, that is not represented in table 2, reported that no routes were aggregated neither in couples nor in triples. That is due to the fact that Dolomitesbus managed to constantly have all the efficient buses required. This metric is derived from the new strategy adopted for 2008, so it is not possible to compare this value to any previous year.

Probe 4, that is not represented in table 2, reported that no complaints were presented from the customers. The quantity of complaints collected by this probe during 2008 cannot be compared to those of 2006/2007 since in these years the company has never stored the complaints received.

The performances measured are the results of the combination of the following factors:

1. The analysis of the maintenance process taken in phase 1;
2. The adoption of significant changes (i) on the company business strategy concerning the main-tenance process, and (ii) on the relating business process;
3. The development of new functionalities having the peculiar characteristic that included the probes for the existing ERP system.

6 Conclusions

This paper addresses the issue of monitoring business process performance in relation to strategic objectives. The solution we proposed is the development of monitoring agents called Probes. Probes are developed combining the technique of BMM and of the GQM. We present the development of Probes by means of a practical case study of a bus company. The company defined the Probes and integrated them into its ERP

System in order to have a real time control of its strategic objectives. According to the company managers of our case study, Probes were able to enhance the overall organization strategy understanding, allowing the managers to take effective action to change inappropriate behaviors over the company processes.

The integration of Probes into a company IT infrastructure require a customization of the ERP System, hence an investment of resources is required by the company. The benefit of developing Probes is that they help managers to better understand the achievement of the strategies they development, but Probes benefits are limited if not combined with proper managers actions and decisions.

One of the future topic of research will be the automation of the Probes. The automation can be done by means of an appropriate ontology reasoner that first analyze the company business strategy written in a controlled natural language and then check if some metrics of the business processes respect the business strategy.

References

1. Artigliano, F., Ceravolo, P., Fugazza, C., Storelli, D.: Business Metrics Discovery by Business Rules. In: Lytras, M.D., Carroll, J.M., Damiani, E., Tennyson, R.D. (eds.) WSKS 2008. LNCS, vol. 5288, pp. 395–402. Springer, Heidelberg (2008)
2. Basili, V., Caldiera, G., Rombach, D.: Goal Question Metric Paradigm. In: Encyclopedia of Software Engineering, vol. 1. John Wiley and Sons, Chichester
3. Basili, V., Heindrich, J., Lindvall, M., Münch, J., Regardie, M., Rombach, D., et al.: Bridging the Gap Between Business Strategy and Software Development. In: International Conference on Information Systems Management, Québec, Canada (2007)
4. Damiani, E., Mulazzani, F., Russo, B., Succi, G.: SAF: Strategic Alignment Framework for Monitoring Organizations. In: 11th International Conference on Business Information Systems. Springer, Heidelberg (2008)
5. Gross, H.G., Melideo, M., Sillitti, A.: Self Certification and Trust in Component Procurement. Journal of Science of Computer Programming 56, 141–156 (2005)
6. Henderson, J.C., Venkatraman, N.: Strategic alignment: Leveraging information technology for transforming organizations. IBM Systems Journal 32(1), 472–484 (1993)
7. Kaplan, R.S., Norton, D.P.: The Balanced Scorecard: Translating Strategy into Action. Harvard Business School, Boston (1996)
8. OMG. Business Motivation Model (BMM) (2006), http://www.omg.org/docs/dtc/06-08-03.pdf
9. Reich, B.H., Benbasat, I.: Measuring the Linkage Between Business and Information Technology Objectives. MIS Quarterly 20(1), 55–81 (1996)
10. Rossi, B., Russo, B., Succi, G.: Evaluation of a Migration to Open Source Software. In: St. Amant, K. (ed.) Handbook of Research on Open Source Software: Technological, Economic, and Social Perspectives, IGI Global, p. 728 (2007)
11. Silvius, G.: Business & IT Alignment in theory and practice. In: Proceedings of the 40th Hawaii International Conference on System Sciences, Washington, DC, USA, p. 211b (2007)

Text Generation for Requirements Validation

Petr Kroha and Manuela Rink

Faculty of Computer Science, TU Chemnitz
Straße der Nationen 62, 09111 Chemnitz, Germany
{petr.kroha,manuela.rink}@informatik.tu-chemnitz.de
http://www.tu-chemnitz.de

Abstract. In this paper, we describe a text generation method used in our novel approach to requirements validation in software engineering by paraphrasing a requirements model expressed in UML by natural language. The basic idea is that after an analyst has specified a UML model based on a requirements description, a text may be automatically generated that describes this model. Thus, users and domain experts are enabled to validate the UML model, which would generally not be possible as most of them do not understand (semi-) formal languages, such as UML. A corresponding text generator has been implemented and examples will be presented.

Keywords: Requirements, Requirements specification, Requirements validation, Text generation, Requirements modeling.

1 Introduction

The most expensive failures in software projects have their roots in requirements specifications. Misunderstanding between analysts, experts, users, and customers is very common.

The process of requirements specification usually runs as follows. Before the modeling phase of the software development can start, we have to acquire and collect requirements. The obtained interviews and studies result in text documents that describe requirements. Using natural language is necessary because a customer would not sign a contract that contains a requirements definition written in some formal notation (e.g. the Z notation).

After these discussions, a requirements specification must be written that describes the functionality and constraints of the new system in a more detailed way. It is usually written as a combination of text and some semi-formal graphical representation given by the used CASE tool. Since software engineers are not specialists in the problem domain, their understanding of the problem is immensely difficult, especially if routine experience cannot be used. It is a known fact [1] that projects completed by the largest software companies implement only about 42 % of the originally proposed features and functions.

We argue that there is a gap between the requirements definition in a natural language and the requirements specification in some semi-formal graphical representation. The analyst's and the user's understanding of the problem are usually more or less different

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 467–478, 2009.
© Springer-Verlag Berlin Heidelberg 2009

when the project starts. The first possible time when the user can validate the analyst's understanding of the problem is when a prototype is used and tested.

In this contribution, we offer a textual refinement of the requirements definition which can be called requirements description. Our tool forces the analyst to complete and explain requirements and to specify the roles of words in the text in the sense of an object-oriented analysis. During this process, a UML model will be built by our tool driven by the analyst's decisions.

This model is used for the synthesis of a text that describes the analyst's understanding of the problem, i.e. a new, model-derived requirements description will automatically be generated. Now, the user has a good chance to read it, understand it and validate it. His/her clarifying comments will be used by the analyst for a new version of the requirements description. The process is repeated until there is a consensus between the analyst and the user. This does not mean that the requirements description is perfect, but some mistakes and misunderstandings are removed.

In this paper we describe the text generation component of our system, i.e. the used methods and achieved results.

We argue that the textual requirements description and its preprocessing by our tool will positively impact the quality and the costs of the developed software systems because it inserts additional feedbacks into the development process.

The rest of the paper is organized as follows. In Section 2 we discuss related work. In Section 3 we briefly introduce the architecture of the new system. In Section 4 the used method is described. A case study illustrating the method is presented in Section 5. The implementation of the system is described in Section 6 and an example in Section 7. Finally, achieved results are discussed and an outlook is given in Section 8.

2 Related Works

Automatically generated texts can be used for many purposes, e.g. error messages, help systems, weather forecast, technical documentation, etc. An overview is given in [2].

In most systems, text generation is based on templates corresponding to model elements. There are rules on how to select and instantiate templates according to the type and contents of an element of the model. String processing is used as a main method. We used it in the first version of our system and found that texts generated in this way were very large and boring. Another disadvantage was that the texts we generated were often not right in the sense of grammar [4]. Building all possible grammatical forms for all possible instantiations had been too complex. Further, the terminology used in the generated texts also included specific terms from the software engineering domain, which reduces text understandability for users and domain experts. The maintenance and evolution of such templates was not easy.

Except for our previous work [5], [6], there are at least two similar approaches with the aim of generating natural language (NL) text from conceptual models in order to enable users to validate the models, which are briefly characterized in turn.

The system proposed by Dalianis [7] accepts a conceptual model as its input. A query interface enables a user to ask questions about the model, which are answered by a text generator. User rules are used for building a dynamic user model in order to select the needed information.

A discourse grammar [8] is used for creating a discourse structure from the chosen information. Further, a surface grammar is used for surface realization, i.e. for the realization of syntactic structures and lexical items.

ModelExplainer [9], [10] is a web-based system that uses object-oriented data modeling (OODM) diagrams as starting point, from which it generates texts and tables in hypertext which may contain additional parts not contained in the model. The result text is corrected and adjusted by the RealPro system [11], [12], which produces sentences in English. However, since NL generation is based on OODM diagrams alone, it is confined to static models. The problems concerning text structure are described in [13].

In the approach given in [14], there is defined a specific Requirement Specification Language and some parsers that can work with it. Textual requirements should be processed automatically. In our approach they are process semi-automatically. The analyst is still an important person and he/she has to understand the semantics of the problem to be solved and implemented.

Because of the disadvantages described above we used a linguistic approach [15] in our last version.

Currently, there are no systems available (we have not even found any experimental systems of that kind) that would follow the idea of using automatically generated textual description of requirements for feedback in modeling. The main application field is in information systems where requirements have to be acquired during an interview then collected, integrated often from many parts together, and processed.

3 Architecture and Dataflow of Our CASE Tool

The main idea of our project TESSI can be seen in Fig.1 where we show how our tool will be applied. In the classic approach, the analyst discusses the problem to be solved with the user. The interviews, observations, and knowledge result into a textual document containing the requirements.

This document represents the analyst's understanding of the problem. It is very likely that the analyst and the user understand some words (some concepts) differently, it is very likely that the user holds some facts for self-evident and thinks they are not worth being mentioned. It is also very likely that some requirements have been forgotten. The document is a starting point to the next analysis. Using our tool TESSI the analyst identifies classes, methods, and attributes in the way how he/she understands the textual requirements and stores them into a UML-model. Our new approach is that from this UML-model a text can be generated that reflects how the analyst modeled the problem. The generated text is given to the user.

The user does not understand the UML-model but he/she can read the text generated and can decide whether it corresponds to his/her wishes. He/she discussed it with the analyst and the next iteration of the process of requirements refinement starts. Additionally, our tool can generate some simple questions, e.g. concerning constrains of attributes. These questions can influence the next iteration text, too.

After some iterations, when the user and the analyst can not find any disproportions, the last UML-model will be exported to the next processing. We use an interface to

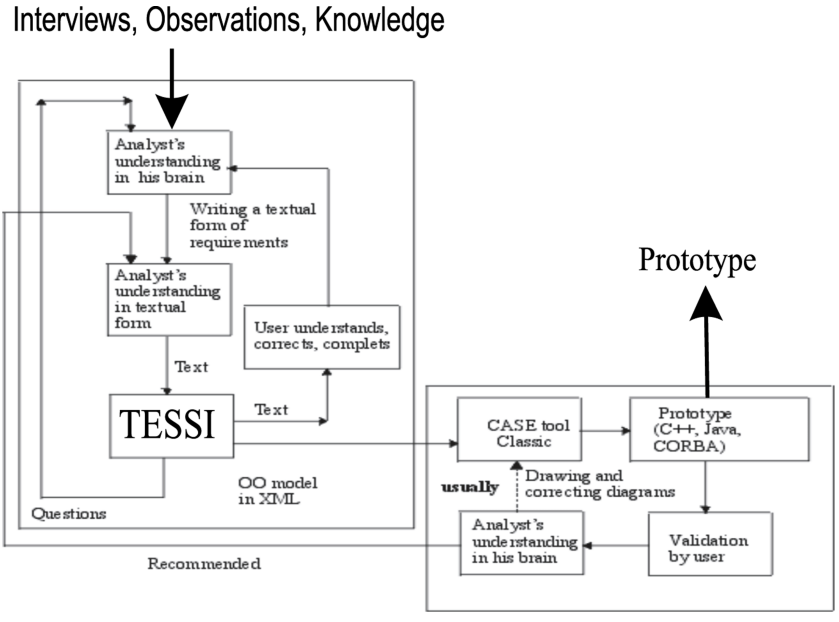


Fig. 1. The TESSI main idea - Data flows and processing

Rational Software Modeler (IBM). This tool produces diagrams of any kind, fragments of code in different programming languages, etc. The fragments of code have to be further completed and developed to a prototype. The prototype will be validated by the user and his/her comments will be inserted into the textual description of requirements.

As we can see our approach means that we use one additional feedback during the modeling before an executable prototype is available. It is very well known that the mistakes from requirements are very expensive because:

- it is expensive to find them because the costs grow up in exponential proportion to the distance between the time point when the mistake occured and the time point when the mistake was corrected,
- it is very likely that parts of the design and programming efford have been invested in vain and these parts have not only to be corrected but they have to be developed again.

The implemented component for text generation is a part of our CASE tool. As mentioned above, in the first phase of requirements acquisition, a text containing knowledge about the features of the system to be developed, is written in cooperation between the analysts, domain experts, and users. The analyst processes this text and, using the MODEL component, decides which parts of the text can be associated to which parts of the UML model. Then the GENERATOR component generates a text corresponding to the UML model and the user validates it. This process can iterate (see Fig. 2) until the differences disappear.

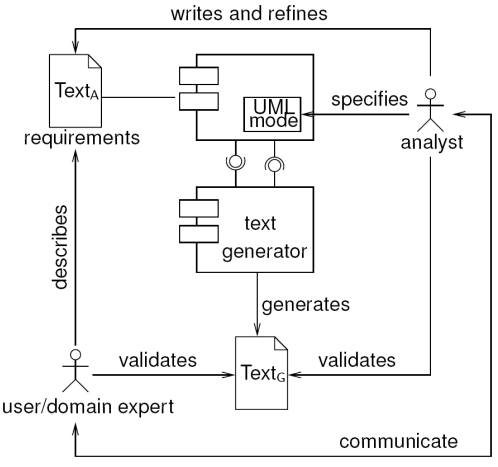


Fig. 2. Architecture of the CASE tool

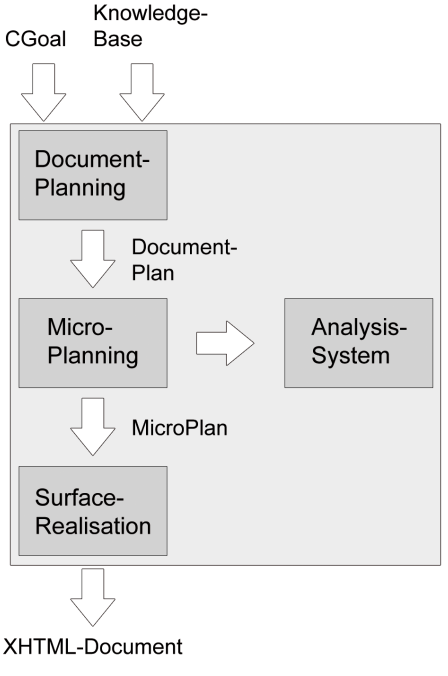


Fig. 3. Architecture of the text generator component

4 Generate Natural Language Text from UML Model

For the purpose of paraphrasing the specified UML model for users and domain experts, an NL text is generated from the UML model.

Differently from works that use templates completed with information from assigned isolated model elements our linguistic approach can collect and combine pieces of information from the whole model for using them together in sentences of the generated text.

For the component GENERATOR we used the standard pipeline architecture [15] for NL generation, extended by an additional module used for NL analysis tasks [16]. Three modules are arranged in a pipeline where each module is responsible for one of the three typical NL generation subtasks, which include, in this order, document planning, micro planning and surface realization (Fig. 3). The output of one module serves as input for the next one.

The input to the document planner is a communicative goal which is to be fulfilled by a text generation process. The communicative goal is basis for the selection of information (content determination) from a knowledge base.

In our case, the goal is to validate a requirements model and the knowledge base is the model itself. Output of the document planner is:

- a document plan,
- a tree structure with message nodes,
- structural nodes.

Message nodes store pieces of information (NL text fragments) to be expressed in a sentence, structural nodes indicate the composition of the text and the order in which the sentences must occur.

The micro planner accepts a document plan as its input and transforms it into a micro plan by processing message nodes. Sentence specifications are produced, which can be either strings or abstract representations describing the underlying syntactic structure of a single sentence. In the latter case this is done by a complex process (as described below) involving the tasks of NL parsing, linguistic representation and aggregation of text fragments as well as choosing additional lexems and referring expressions (articles).

A micro plan is transformed into the actual text (surface text) of a certain target language by a surface realizer. During structural realization the output format of the text is developed. The process of linguistic realization performs the verbalization of abstract syntactic structures by determining word order, adding function words and adapting the morphological features of lexems (e.g. endings).

4.1 The Approach

First, we wrote the presupposed text that should be generated in our case study using semantic relations between its parts, which can be derived from the UML model. There are semantic relations in UML models between the following elements:

- use case and sequence diagram
- class and state machine
- use case and transition in a state machine

Examples are given in Section 5.

After this, we analyzed possibilities to derive the target text from an existing UML model. We found that there are:

- fixed text fragments that specify the structure of the generated text
- directly derivable text fragments that can be copied, e.g. names of classes
- indirectly derivable text fragments that depend on syntax and morphology rules
- not derivable text fragments that cannot be derived from the model

We noticed that a minor part of the text could be produced by a simple template-based approach. This is the case for sentences combining fixed text fragments and directly derivable text fragments.

To simplify the generation process where possible we made our text generator capable to perform template-based generation as well. However, for the major part of the text a generation based on templates was not sufficient. In cases where sentences contain indirectly derivable text fragments a method depending on linguistic knowledge was needed.

4.2 Document Planning

In the stage of document planning text fragments and additional information are collected in several message types. A message type reflects a certain type of sentence defining its underlying informational structure. Thus, each message type defines a set of individual attribute-value pairs that contribute to the content of the sentence. An example is given below.

$$\begin{bmatrix} \textbf{type:} & \text{InteractionMsg} \\ \textbf{message:} & \text{“specify search criteria”} \\ \textbf{sender:} & \begin{bmatrix} \textbf{body:} & \text{“user”} \\ \textbf{isActor:} & \text{true} \end{bmatrix} \\ \textbf{receiver:} & \begin{bmatrix} \textbf{body:} & \text{“Media Admin.”} \\ \textbf{isActor:} & \text{false} \end{bmatrix} \end{bmatrix}$$

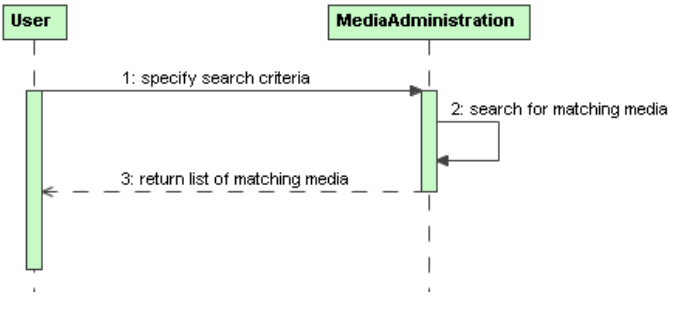


Fig. 4. Sequence diagram

4.3 Mirco Planning

During the stage of micro planning these messages are at first subject to a preprocessing step. The text fragments stored in the message are given to a NL parser which produces amongst others typed dependency structures for syntactic representation.

After preprocessing, a message is replaced by a proto sentence specification. A proto sentence specification structurally corresponds to a certain message type. The difference is that all textual values are replaced by dependency structures of the represented text fragments.

In the next step, the parser-generated dependency structures stored in a proto sentence specification are transformed into single DSyntS [12] instances. A DSyntS is a special type of dependency structure which differs according to the NL parser that was used more or less from the source dependency structure. Differences may exist in the word pairs that are related, in the types of dependency relations that may occur and in additional features used to specify a word.

Now that we have DSyntS instances for all of the text fragments which are to be combined in a sentence, these instances can be merged into a DSyntS for the entire sentence. This is done by means of tree operations. As the input text fragments do not include all the lexems that are necessary to build up the whole sentence, additional lexems have to be determined, represented by new DSyntS-nodes and included in the sentence DSyntS. Furthermore, dependency types which specify the edges of the DSyntS-tree are modified and features such as tense, number, word position or type of article are added to the DSyntS-nodes. The result is a sentence specification that can be given to the surface realizer.

5 Case Study

To illustrate our text generation method, we now apply it to a contrived specification of a library automation system.

As an example, we combine information from use case diagrams and state machine diagrams in the following way:

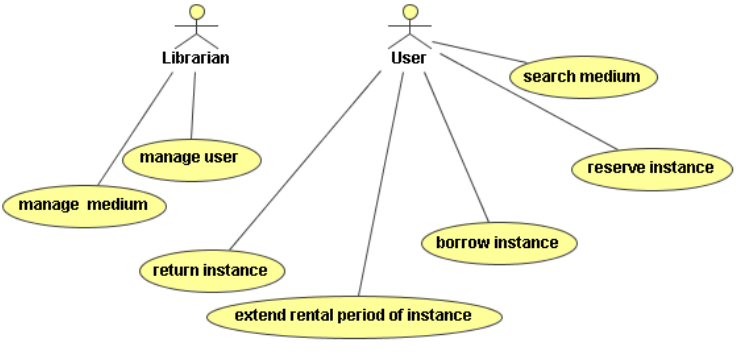


Fig. 5. Use case diagram

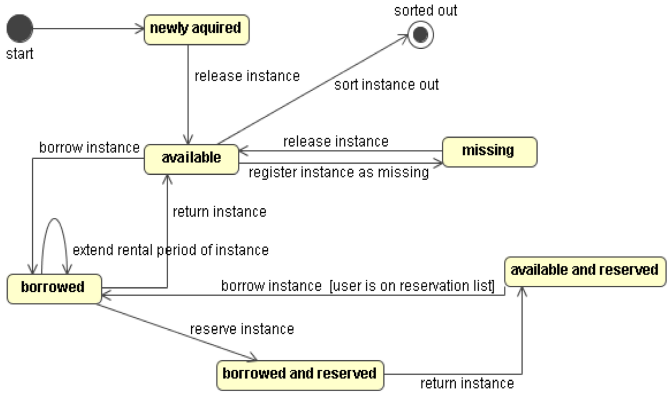


Fig. 6. State machine diagram

- Use case diagram ... “borrow instance”
- State machine diagram ... “available”, “available and reserved”
- Generated text:
  “Only instances can be borrowed which are available, or available and reserved and the user is the first on the reservation list”.
  “If the instance is signed as available the user can borrow the instance. The instance will afterwards be signed as borrowed. Alternatively, if the instance is signed as available and reserved and the user is the first on the reservation list the user can borrow the instance. The instance will afterwards be signed as borrowed.”

6 Implementation

The current system is a Java/Eclipse application. The NL generation component has been developed as a module and integrated into the system as a plug-in component.

The generator produces output texts in the English language. The results of the different generation steps (document plan, micro plan, output text) are represented using XML technology.

The task of document planning is managed by schemata, each of them is responsible for the generation of a certain part of the document plan. To fulfill the communicative goal Describe_Dynamic_Model several schemata exist which specify the interwoven steps of content determination and document structuring.

According to the two main subtasks the micro planner performs, the module includes a ProtoSentenceSpecBuilder and a SentenceSpecBuilder. The ProtoSentenceSpecBuilder processes an input message and produces a proto sentence specification. After that, the SentenceSpecBuilder transforms the proto sentence specification into a sentence specification. The SentenceSpecBuilder provides an interface that is realized by several components according to the different types a proto sentence specification may have. Thus, the individual implementations encapsulate the complete knowledge

needed for the creation of the DSyntS of a specific sentence type from data stored in the proto sentence specification.

For NL analysis the Stanford parser [17], [18] is used. This parser provides two different parsing strategies (lexicalized factored-model, unlexicalized PCFG) that both can be chosen for the task of preprocessing (micro planning). Access to the parser and corresponding components used for the processing of dependency structures and DSyntS is granted by the interface the AnalysisSystem provides.

Our generator component produces output texts formatted in XHTML. The mark up is developed in the stage of structural realization performed by the XHTMLRealiser. To accomplish linguistic realization and produce the surface form of the output texts RealPro [12] is used.

7 Example of the Text Generated for Validation

As an example we show a fragment of a generated text that is a part of a generated Library system description, i.e. the text is generated from UML-model of a Library system. In the following there is the description of the function BorrowInstance:
BorrowInstance

This function can be done by a user.

Preconditions and Effects:

If the instance is signed as available the user can borrow the instance. The instance will afterwards be signed as borrowed. Alternatively, if the instance is signed as available and reserved and the user is on the reservation list the user can borrow the instance. The instance will afterwards be signed as borrowed.

Procedure:

1. The user identifies himself.
2. The user specifies the instance by the shelfmark.
3. A component (User Administration part of the Library system) registers the instance in the borrowed-list of the user account.
4. A component (Media Administration part of the Library system) registers the user in the borrowed-list of the instance.
5. A component (Media Administration part of the Library system) changes the status of the instance.
6. A component (Media Administration part of the Library system) returns the receipt.

8 Achieved Results and Conclusions

A component has been designed and implemented which serves as an important basis for sophisticated NL text generation with the purpose of validating requirements analysis models. The text generator performs text generation in three consecutive steps: document planning, micro planning and surface realization. It presents an approach

to text generation based on textual input data using NL-analysis- and NL-generation-techniques. Compared to texts produced by the pre-existing template-based text generator, texts generated by the new non-trivial text generator are definitely more structured, more clearly arranged and more readable.

Further, the vocabulary used should be more understandable for people outside the software industry. As far as it has been considered possible, generated texts do not contain terms specific to software engineering. Due to the usage of RealPro for surface realization, the grammar of generated sentences is also more correct than before.

Currently, the text generator is capable of producing NL texts from use cases, sequence diagrams and state machines. As the architecture has been designed with the aim of easy extensibility, it should not be too difficult to integrate text generation functionality for other UML model elements as well. Furthermore, it is possible to adapt the text generator to other target languages.

A number of open issues may be addressed in the future: prevention of generation errors caused by the NL parser, improvement of the micro planner, integration of text schemata for other model elements (such as static structures like classes).

Further it is desirable to evaluate our proposed validation approach by application to real-world projects. This is not easy because it is necessary to persuade the management in a software house to run a project in two teams (one team should use our tool) and then to compare the results.

References

1. Chaos report. The Standish Group (1995)
2. Paiva, D.: A survey of applied natural language generation systems (1998), http://citeseer.ist.psu.edu/paiva98survey.html
3. Hoppenbrouwers, J., van der Vos, A.J., Hoppenbrouwers, S.: NL structures and Conceptual modelling: the KISS case. In: Proceedings of the 2nd. International Workshop on the application of Natural Language to Information Systems (NLDB 1996). IOS Press, Amsterdam (1996), http://citeseer.nj.nec.com/hoppenbrouwers96nl.html
4. Kroha, P., Strauß, M.: Requirements specification iteratively combined with reverse engineering. In: Plasil, F., Jeffery, K.G. (eds.) SOFSEM 1997. LNCS, vol. 1338. Springer, Heidelberg (1997)
5. Kroha, P.: Preprocessing of requirements specfication. In: Ibrahim, M., Küng, J., Revell, N. (eds.) DEXA 2000. LNCS, vol. 1873, p. 675. Springer, Heidelberg (2000)
6. Rink, M.: Text generator implementation for description of UML 2 models. MSc. Thesis, TU Chemnitz (2008) (in German)
7. Dalianis, H.: A method for validating a conceptual model by natural language discourse generation. In: Loucopoulos, P. (ed.) CAiSE 1992. LNCS, vol. 593, pp. 425–444. Springer, Heidelberg (1992)
8. Hovy, E.: Automated discourse generation using discourse structure relations. Artificial Intelligence, Band 63(1-2) (1993)
9. Lavoie, B., Rambow, O., Reiter, E.: Modelexplainer - cogentex - generating textual descriptions of object-oriented data models (1996), http://www.cogentex.com/research/modex/index.shtml
10. Lavoie, B., Rambow, O., Reiter, E.: Customizable descriptions of object-oriented models. In: Proceedings of the Fifth Conference on Applied Natural Language Processing, pp. 253–256. Morgan Kaufmann Publishers, Washington (1997)

11. Lavoie, B., Rambow, O.: A fast and portable realizer for text generation systems. In: Proceedings of the 5th Conference on Applied Natural Language Processing, pp. 265–268. Morgan Kaufmann Publishers, San Francisco (1997)
12. Realpro: General english grammar, user manual (2000), http://www.cogentex.com/papers/realpro-manual.pdf
13. Mann, W.: Text generation: The problem of text structure. In: McDonald, D.D., Bolc, L. (eds.) Natural Language Generation Systems, pp. 47–68. Springer, Heidelberg (1988)
14. Videira, C., Ferreira, D., da Silva, A.: A linguistic patterns approach for requirements specification. In: Proceedings of the 32nd EUROMICRO Conference on Software Engineering and Advanced Applications (EUROMICRO-SEAA 2006) (2006)
15. Reiter, E., Dale, R.: Building natural language generation systems. Studies in Natural Language Processing, Journal of Natural Language Engineering 3(1), 57–87 (1997)
16. DeSmedt, K., Horacek, H., Zock, M.: Architectures for natural language generation: Problems and perspectives. In: Adorni, G., Zock, M. (eds.) EWNLG 1993. LNCS, vol. 1036, pp. 17–46. Springer, Heidelberg (1996)
17. Klein, D., Manning, C.: Fast exact inference with a factored model for natural language parsing. In: NIPS, vol. 15. MIT Press, Cambridge (2003)
18. Klein, D., Manning, C.: Accurate unlexicalized parsing. In: Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics (ACL 2003), pp. 423–430 (2003)

Automatic Compositional Verification of Business Processes

Luis E. Mendoza[1] and Manuel I. Capel[2]

[1] Processes and Systems Department, Simón Bolívar University
P.O. Box 89000, Baruta, Caracas 1080–A, Venezuela
lmendoza@usb.ve
http://www.lisi.usb.ve
[2] Software Engineering Department, University of Granada
ETSI Informatics and Telecommunication, 18071 Granada, Spain
manuelcapel@ugr.es
http://lsi.ugr.es/~mcapel/

Abstract. Nowadays the *Business Process Modelling Notation* (BPMN) has become a standard to provide a notation readily understandable by all business process (BP) stakeholders when it comes to carrying out the *Business Process Modelling* (BPM) activity. In this paper, we present a new *Formal Compositional Verification Approach* (FCVA), based on the *Model–Checking* verification technique for software, integrated with a formal software design method called MEDISTAM–RT. Both are used to facilitate the development of the *Task Model* (TM) associated to a BP design. MEDISTAM–RT uses UML–RT as its graphical modelling notation and CSP+T formal specification language for temporal annotations. The application of FCVA is aimed at guaranteeing the correctness of the TM with respect to initial property specification derived from the BP rules. One instance of a BPM enterprise–project related to the *Customer Relationship Management* (CRM) business is discussed in order to show a practical use of our proposal.

Keywords: Business Process Modelling, Verification, Model–Checking, Task Model, Formal Methods.

1 Introduction

Business Process (BP) is commonly understood as the way companies carry out their business objectives or *user goals*. *Business Process Modelling Notation* (BPMN) [1] is the new standard for modelling BPs and web services processes, as it is put forth by the *Business Process Management Initiative* (BPMI —*www.BPMI.org*). BPMN consists of one diagram, called *Business Process Diagram* (BPD), which is based on a flowcharting technique tailored for creating graphical models of BPs operations. Then, a BPD is built as a network of graphical objects, which represent activities (i.e., business tasks) and control defining their execution order. This diagram has been designed to be easy to use and understand, but it also provides the ability of modelling complex business [1].

A *Business Process Task Model* (BPTM) can be defined as the set of logical descriptions of business tasks needed to be carried out in order to achieve the user goals

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 479–490, 2009.
© Springer-Verlag Berlin Heidelberg 2009

[2]. *Business Process Modelling* (BPM) is a "too rudimentary" activity up to now, since still there is a lack of maturity at the level of soundness and descriptive power of current methods and languages for BPTM. Therefore, different methods have been proposed in the literature to guide the development of a TM through the different life cycle phases (see [3,4]). Nevertheless, still there is not a standard method or commercially available tool currently able to tackle all the requirements of the different tasks being modelled.

In order to establish some criteria to make a selection of BPTM methods, a number of requirements are described in the *Web Services Business Process Execution Language* (WSBPEL) [5] specification. Among others, our approach accomplishes the following WSBPEL requirements: it uses "specification" languages defined for describing models (such as a subset of UML–RT and CSP+T), which allows for reaching a high level of detail necessary to specify BPs' behaviour at different development stages as well as performing model evaluation, and it allows an automatic analysis/design support provided by software tools. Taking into account these requirements, and with the idea of obtaining a directly executable/verifiable model from a conceptual[1] one, is that we have proposed a *Formal Compositional Verification Approach* (FCVA) in which the *correctness* of BPTMs design can be *model–checked* against the required *Task Model* (TM) properties and BP rules. Prior to start the verification activity, we should build and specify the BPTM in a formal language because we do not obtain it as result of the BPMN application. In this sense, MEDISTAM–RT (Spanish acronym for *'Systematic Design Method Based on Model Transformations for Real–Time systems'*) [6] is used to facilitate the identification of all the design issues in BPTM, such as its behavioral aspects and temporal constraints. An instance of the *Customer Relationship Management* (CRM) case study is discussed to show how these issues are addressed.

The remainder of this paper is structured as it follows. In the next section we give a brief description of the concepts related with BPs and BPTMs specification. Afterwards, we describe our proposal in detail. Then, we apply our proposal to a BPM case study related to the CRM business. The last section gives a conclusion and discusses future work.

2 BP and BPTM Specification

The most basic BP model for BPM consists of a sequence of *tasks* [3] performed in order to achieve a goal. A more general TM structure is a typed hierarchy [3] of tasks, representing a large number of possible real–world scenarios shown in a compact form. A *scenario* describes how the workflow of a particular BP is realized within the business model, in terms of collaborating business objects [7]. Different scenarios could be generated by choosing different combinations of parallel subtasks (if they are not all mandatory), or by executing them in a different order from the first one observed. Also the scenarios represent different viewpoints regarding how the BP will apply to the participants in the BP [1].

According to BPMN guidelines, in order to model a BP flow, you simply model the events that occur to start a process, the processes carried out, and the outcome of the

[1] A descriptive model of a BP based on qualitative assumptions about its elements, their interrelationships, and BP boundaries.

process flow, by using a single business process diagram called BPD. The events and processes are placed into shaded areas called *pools* to denote that the participants are carrying out a process. You can further partition a pool into *lanes*. A pool typically represents an *organization* and a lane typically represents a *department* or *business worker* within that organization (although you may make them represent other things such as functions, applications, and systems). Both, pools and lanes, represent *participants in a process*. In summary, a BPD represents a scenario of a business model.

On the other hand, to make a BP being supported by *Information Technology* (e.g., *Workflow Management Systems* (WFMS) [8]), business and software analysts will require a higher level of accuracy in the BPTM. Thus, the *tasks*, i.e., the atomic activities included in a process [1] and the BPTM are both required to develop the end–user and/or applications necessary to carry out the BP tasks. Also it is needed the behavioural verification of a BPTM with respect to non-functional requirements, such as *Quality of Service* (QoS), task timeliness, performance, etc. These requirements do not depend on the results yielded by the BPTM realization only, but also on the execution time of tasks. For instance, the correctness of the BPTM depends on the verification of real–time tasks always meet their deadlines. A formal verification of BPTM that includes checking all non–functional requirements, such as real–time constraints, is mandatory in the development/modelling of critical BPs. Moreover, validation of results is extremely expensive and risky for the development process when its application is postponed until the system deployment phase.

Our aim here is supporting the entire modelling cycle, thus we need a comprehensive, behavioural–oriented, object–based approach that can be used to support the BPM from the beginning of the life cycle's design stage. In this work, we are using MEDISTAM–RT for obtaining the TM design of a given BP. MEDISTAM–RT is a combined method that incorporates *UML Real–Time* (UML–RT) for TM modelling and *Communicating Sequential Processes + Time* (CSP+T) formal language to specify and deal with time at the model level. Object Oriented design features of UML–RT can be used to structure BPTM, thus offering an ease way to analyze, design, compose, and reuse tasks submodels, and hence supporting the flexibility required by BPs. MEDISTAM–RT starts from the UML–RT model and allows, by following a top–down strategy, the integration of UML *Timed State–Machines* (TSMs) and composite structure diagrams in order to completely describe the behaviour of the BP participants and the specification of real–time constraints defined on BPs. This transformation is carried out within a common methodological framework given by the UML notation. In a more technical way, we can say that MEDISTAM–RT is a systematic transformation procedure for obtaining the complete specification of a real–time system (in this case applied to BPTM) by giving structured operational semantics, in terms of CSP+T process algebra, to the semiformal UML–RT analysis entities. This result is obtained in the second phase, by applying a bottom-up strategy. The transformation of UML–RT diagrams of BPTM components into CSP+T processes is carried out by a system of rules [6], so that the final BPTM design is correct by construction. Mapping links are continuously established between the UML–RT diagrams of components in which the BPTM is structured and their formal specifications in terms of CSP+T processes. These links aim to signify how CSP+T syntactical terms are used to represent the real–time constraints and the

internal components and connectors that constitute the BPTM architecture, at different levels of description detail, during the entire transformation process. Then, the MC of the BPTM is a result of the verification of the satisfaction relation between CSP+T process terms at different levels of refinement.

CSP+T can be considered a formal specification language of use in BPM. This notation adds to *Timed Communicating Sequential Processes* (TCSP) algebra [9] the ability to deal with temporal restrictions, which allow the specification and treatment of time aspects in the BPTM specification, which thereby can take advantage of all the CSP strengths [9,10] for specifying reactive and interactive systems. CSP+T is a superset of *Communicating Sequential Processes* (CSP), as a major change to the latter, the traces of events are now *pairs* denoted as *t.e*, where t is the global $absolute$ time at which event e is observed. The operators, related with timing and enabling–intervals included in CSP+T are [10]: (a) the special process instantiation event denoted $\star$ (star); (b) the time capture operator ($\bowtie$) associated to the time stamp function $a_e = s(e)$ that allows storing in a variable a (marker variable) the occurrence time of an event e (marker event) when it occurs; and (c) the event–enabling interval $\mathrm{I}(T, t_1).a$, viewed as representing timed refinements of the *untimed* system behaviour and facilitates the specification and proof of temporal system properties [10]. Then, a good integration of these advances into a method to support the modelling and verification of the TM associated to a BP is considered of great importance in the BPM field. The use of CSP+T allows the use of state–of–the–art CSP verification tools (i.e., *Failures–Divergence Refinement 2* (FDR2) [11]) in all process development steps, thus enabling continuous quality checking of the design. In this way, we will help those responsible for the BPM in checking whether the design fulfills the original property specifications or not. More formally, we can say that our proposal may be considered a way to verify that the TM accurately specified corresponds to a given BP.

3 Verification Approach

The BP model can have several views and each view is expressed through one or more diagrams [12]. The diagrams capture BP rules, goals, relations between business objects and their interactions, resources, events, activities, and so on. When together, these views create a complete BP model. According to BPMN [1] and our objectives, we started from the BPD because is the mechanism used by BPMN for creating BP models, while at the same time BPD is able to handle the complexity inherent to BPs [1].

As we said in a previous section, a TM structure is a hierarchy of tasks, representing a large number of possible real–world scenarios expressed in compact form. Thus, we are focused here on the BPTM and the set of overlapping *user scenarios*, which allow us to obtain a description of most of the activities that a BPTM must take into account [4]. However, a complete behavioural description of the BP cannot be obtained by only using the information provided by TM without considering individual behaviour of the *Participants in the BP* (PBPs). As result we obtain a set of detailed TSMs which complete the behavioural description of the PBPs that perform the tasks described by the BPTM. We can check the correctness of the BPTM by using a *Model–Checking* (MC) tool w.r.t. previously specified properties of the BP. Our FCVA is based on the main principle of

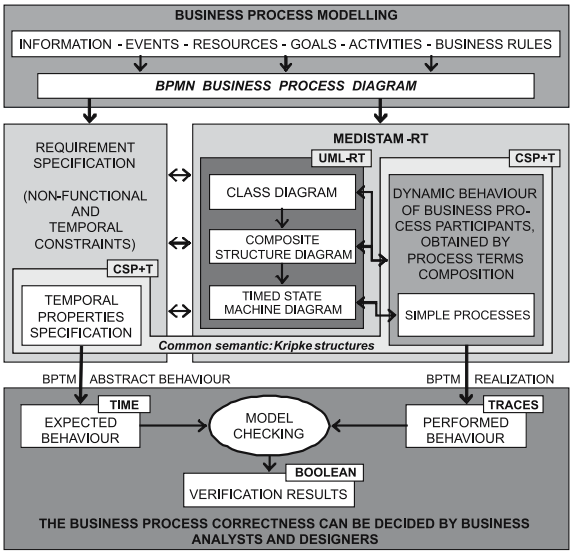


Fig. 1. Integrated view of our FCVA proposal

MC [13,14] to verify the *correctness* of a BPTM, i.e., checking that a model of the design (i.e., the BPTM) *does not violates* a given property (i.e., a business rule).

On Fig. 1 we see the graphical summary of our proposal that shows different paths to be followed in its application and artifacts (denoted inside brackets) that are obtained from the activities execution. MC concepts are integrated with MEDISTAM–RT, so as to carry out the verification of a BPTM.

To perform the behavioural verification of a BPTM following the standard MC procedure, we need its *realization*, which describes how the PBPs perform the tasks in terms of collaborating workers, and the formal specification of its *properties* representing the characteristics declaration of the BPTM. Both realization and properties specification should be oriented by the associated BP. Afterwards, with a MC tool, we automatically check whether the realization fulfills the BPTM specification.

The complete description of the BPTM behaviour can be obtained as result of applying MEDISTAM–RT to a given BP. A series of BPTM views represented by UML–RT class, composition structure, and TSM diagrams are obtained through following the steps of MEDISTAM–RT. Afterwards, these views are transformed into CSP+T process terms, which share refinement and satisfaction relationships. The work in [15,16] gives further insight on these.

In the same way, some non–functional requirements (i.e., deadlock–freeness, reliability) and temporal constraints (i.e., timeliness, deadlines) that the BPTM must fulfill are modelled by using TSMs; and then transformed into CSP+T process terms by applying a set of rules [6]. By means of CSP+T process terms, these transformation rules describe behaviours equivalent to the one of UML–RT notational elements. In this sense, the verification carried out here exclusively refers to the BPTM behaviour specified by the CSP+T process terms obtained from transformations.

Once obtained the CSP+T process terms, we can proceed to BPTM verification according to the semantics given by KS. By using MC tools it is possible to check whether a BPTM realization in terms of CSP+T processes *satisfies* the expected temporal behaviour of the BPTM in terms of CSP+T processes. Then, the correctness criteria of our approach for MC is the verification of the *satisfaction/refinement* relation between the CSP+T process terms. We obtain the verification of the BPTM realization by the interpretation of boolean expressions (*True*, *False*), according to its *Expected Behaviour Properties* (*EBP*).

As the TM behaviour realization is the combination of each one of its *PBPs Behaviour* (*PBPB*), i.e., the parallel composition of the *PBPB* CSP+T process terms, it is formally possible to assure and obtain the complete verification of the TM realization by using the relation[2]:

$$\|_{i:1\ldots n} PBPB_i \models \bigwedge_{j:1\ldots m} EBP_j \; ; \tag{1}$$

i.e., the parallel composition of the *PBPs Behaviour* satisfy the conjunction of the *Expected Behaviour Properties*.

Since our approach is aimed at representing BPTM concurrent aspects, the contribution is more focused on compositional verification of consistency and synchronization of concurrent tasks which exist in BPTM than in other BPs oriented validations; i.e., according to our approach, the verification of structured process terms can be carried out with correctness by only starting from the verification of the simplest CSP+T process. In MEDISTAM–RT, by encapsulation of its internal behaviour, a component is designed in such a way that only the behaviour in the interface is visible to its environment. This observable behaviour is designed according to the communication protocols (previously) defined to carry out the interactions with other system's components. As a consequence, any execution error due to internal events cannot affect the observable behaviour of the component, thus preventing the component from engaging in any occurrence of non–anticipated and unwanted communications. The latter characteristic guarantees that with MEDISTAM–RT any derived component will be composable [17]. These statements are proved in [6].

4 Case Study

We apply our proposal to one instance of a BPM enterprise–project related to the CRM business. CRM is a strategy that tries to establish and maintain a relationship between a company and its customers [18]. CRM is considered a complex combination of business and technical factors that should be aligned by following a strategy [18].

In our approach, the business requirement analysis and context should have been obtained, and modelled with a set of BPMN BPDs prior to performing the verification of the TM associated to the CRM BP. In summary, the obtained set of BPMN BPDs will include the *Informing Customer*, *Customizing Service*, *Studying behaviour*

[2] The symbol || denotes *parallel composition*, the symbol ⊨ denotes *satisfaction*, and the symbol ∧ denotes *conjunction*.

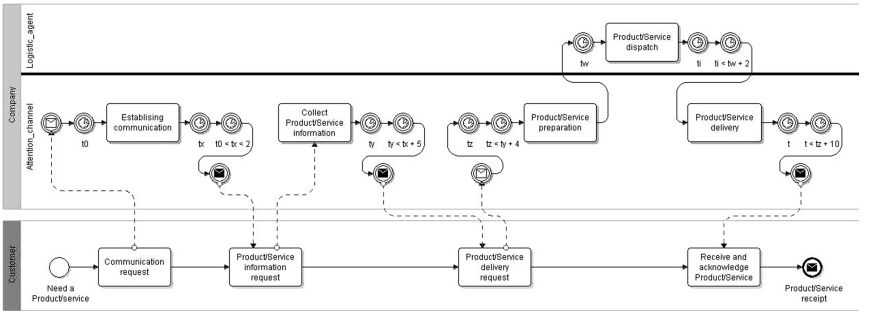


Fig. 2. BPD of the *Product/service Sell* BP

Pattern, *Producing/Providing Service*, *Product/Service Sell*, and *Assisting Customers* BPs that represent a minimum functionality of the CRM strategy and are the key for understanding the CRM business. As our objectives are not to show how the BPM was performed using BPMN, we will only use the BPMN BPD considered of interest to show the verification activity.

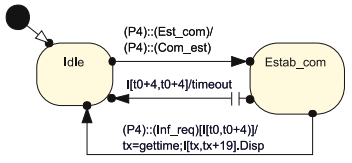
Because of space limitations, we mainly focus on the verification of a part of the TM associated to CRM BP. We selected to work with the *Product/Service Sell* BP, due to its importance to the CRM strategy. The required information to perform the BPTM verification is displayed by the *Product/Service Sell* BPD shown in Fig. 2. As you can see in Fig. 2, the BPD already have time annotations. These annotations correspond with the times brought up in the *Product/Service Sell* BP QoS level agreement, according to the business rules.

4.1 Expected Behaviour Specification

A set of properties were defined to specify the TM associated to *Product/Service Sell* BP expected behaviour. These properties represent the business rules that a BPTM must satisfy to achieve the *Product/Service Sell* BP QoS. Some of these rules are the following ones, (a) BPTM execution deadline, (b) BPTM task execution conformance with respect to a predefined order, (c) *Product/Service Sell* BP safety rules preservation, (d) any product/service request will be finally satisfied by the BPTM, (e) PBPs will never be unsynchronized, and (f) product/service will not ever be dispatched by the BPTM without receiving a previous product/service request.

Considering that we are conducting the application of our approach to a simplified representation of the TM associated to *Product/Service Sell* BP, we decided to focus on assuring the correctness of the synchronization between the *Attention Channel* and *Logistic* agents (PBPs realization) only. The decision was made after performing an analysis of the accompanying particular requirements. Some other requirements, such as meeting BPTM execution deadline, and respecting the execution times of the tasks established by the TM associated to *Product/Service Sell* BP, have been taken into account.

Taking the *Product/Service Sell* BPMN BPD as initial input and the BPTM properties indicated previously, we first define “what” we expected of the process to do when



(a) TSM

$$
\begin{aligned}
ESP_P_S_sell &= \star.t_0 \rightarrow Idle \\
Idle &= (P4?Est_com \rightarrow (P4!Com_est \rightarrow \\
&\quad Estab_com) \\
Estab_com &= ((P?I[t0, t0+4).Inf_req \rightarrow \\
&\quad (tx = gettime \rightarrow \\
&\quad I[tx, tx+19].Disp \rightarrow Idle)) \\
&\quad | \\
&\quad (I[t0+4, t0+4] \rightarrow timeout \rightarrow \\
&\quad Idle) \\
V_{P4}(CSP + T(P_S_sell)) &= \{Est_com, Com_est, Inf_req, Disp\}
\end{aligned}
$$

(b) CSP+T term

Fig. 3. Abstract *Product/Service Sell* TM and specification

it receives a specific request from the Customer, this view is modelled by the TSM (see Fig. 3 (a)) describing the abstract behaviour that should be satisfied by the BPTM realization.

This TSM diagram shows the main states representing the expected abstract behaviour, which BPTM realization must satisfy to accomplish the above described properties. As you can see in Fig. 3 (a), time annotations depicted on the transitions should be consistent with time annotations made in the BPMN BPD shown in Fig. 2. The former ones represents the time constraints, or the maximum elapsed time, within which the *Attention Channel* and *Logistic* agents, i.e. PBPs, must mutually engage and synchronize to fulfil the deadline.

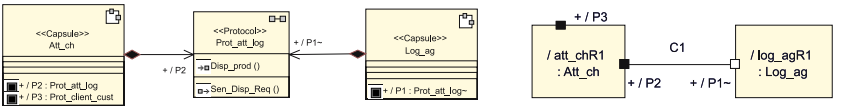
On other hand, events and actions associated to BPD flow objects represent messages (elements of the CSP+T process communication alphabet) exchanged by PBPs to perform the needed collaboration between them to satisfy their execution interactions.

Once obtained the BPTM abstract behaviour, modelled by the called TSM diagram, we use the set of TSM extension rules [6] to obtain CSP+T process term shown in Fig. 3 (b), which also represents it in a textual form. The latter one will be used in the last task of our verification activity as the formal description of the expected behaviour of BPTM whose realization must satisfy. The CSP+T term shown in Fig. 3 (b) (*ESP_P_S_sell*) corresponds to the specification of each of its states, i.e., *Idle* y *Estab_com*, described by the TSM and the relation among them. Each state specification gives a complete description of the message exchange performed between the PBPs and the order in which they must occur to carry out BPTM's tasks execution.

4.2 TM Realization

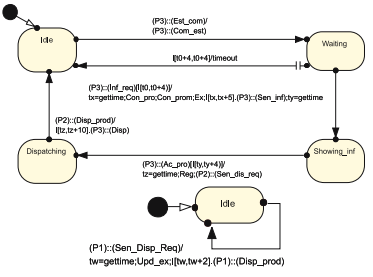
First of all, we design a *P_S_sell* UML–RT class diagram as shown in Fig. 4 (a) which models the PBPs (*Att_ch* and *Log_ag*) as capsules and represents it in form of protocols which includes the set of communications which these elements must exchange to perform the tasks. The interfaces (ports) from which they communicate are also specified. Afterwards, in order to give a complementary view of how these elements are connected, we design a composite structure diagram as indicated in MEDISTAM–RT and shown in Fig. 4 (b). The UML–RT class and composite structure diagrams were designed to prepare a basis to obtain a consistent communication between the PBPs (i.e., a sequence of events that models how the task goals can be obtained by the PBPs).

To obtain the complete specification of the BPTM realization behaviour, the individual behaviour modelled and specified of the PBPs (*Att_ch* and *Log_ag*) can be



(a) Class diagram (b) Composite structure diagram

Fig. 4. UML–RT architecture of *Product/Service Sell* BPTM realization



(a) TSMs

$$
\begin{aligned}
P_S_sell &= \mathrm{CSP} + \mathrm{T}_{P2,P1}(Att_Ch, C1, Log_Ag) \\
&\quad V_{P2}(CSP + T(Att_Ch))||[P2_{in}, P2_{out}]|C1 \\
&\quad ||[P1_{in}, P1_{out}]|V_{P1}(CSP + T(Log_Ag)) \\
Att_Ch &= \star.t_0 \rightarrow Idle \\
Idle &= (P3?Est_com \rightarrow (P3!Com_est \rightarrow \\
&\quad Waiting) \\
Waiting &= (I[t0, t0+4].P3?Inf_req \rightarrow \\
&\quad P!I[tx, tx+5].Sen_inf \rightarrow \\
&\quad Showing_inf)|(Con_pro \rightarrow Ex \rightarrow \\
&\quad (I[t0+4, t0+4] \rightarrow timeout \rightarrow Idle) \\
Show_inf &= (P3?Ac_pro \rightarrow (register \rightarrow \\
&\quad P3!Sen_Disp_Req \rightarrow Dispatching) \\
Dispatching &= (P3?Disp_prod \rightarrow P3!Idle) \\
V_{P3}(CSP + T(Att_Ch)) &= \{Est_com, Com_est, Inf_req, \\
&\quad Sen_inf, Ac_pro\} \\
V_{P2}(CSP + T(Att_Ch)) &= \{Sen_dis_req, Disp_prod\} \\
Log_Ag &= \star.t_0 \rightarrow Idle \\
Idle &= P1?Sen_dis_req \rightarrow \\
&\quad (update \rightarrow P1!Disp_prod \rightarrow Idle) \\
V_{P2}(CSP + T(Log_Ag)) &= \{Sen_dis_req, Disp_prod\}
\end{aligned}
$$

(b) CSP+T terms

Fig. 5. PBPs' *Product/Service Sell* BPTM realization and specification

coordinated and synchronized, in such a way that working together they can provide the product/service expected. We therefore design a TSM for each PBP conforming to the protocols defined previously. Then, we obtain the TSMs as showed in Fig. 5 (a).

Finally, we use the TSM extension rules explained in [6] to obtain the BPTM realization behaviour CSP+T process terms show in Fig. 5 (b), which specify the PBPs' behaviour (subcapsules *Att_ch* and *Log_ag*), involved in the execution of the BPTM realization. These CSP+T process terms specifies the BPTM realization behaviour modelled by the TSM shown in Fig. 5 (a), and will be used in the last task of our verification activity as the BPTM realization that must be fulfill the expected BPTM behaviour specified in Fig. 3 (b).

4.3 TM Verification

Once obtained the required CSP+T process terms, we execute the verification of the BPTM realization. To carry out the verification we check if the CSP+T process P_S_sell ($[Att_ch \| Log_ag] \backslash$C1) satisfies the $ESP_P_S_sell$ specification. In prior works [16,15] formal proofs have been carried out to show how to perform the verification of CSP+T process terms.

From the specification in Fig. 3 (b) and the realization in Fig. 5 (b), we proceed to obtain their verification. In this case, we need to take into account that we are working with a simplified representation of the TM associated to CRM BP. The FDR2 MC tool [11] was used to carry out the verification. As it can be observed in Fig. 6, the execution of the verification software concludes (notice the check–mark at the left of the dark

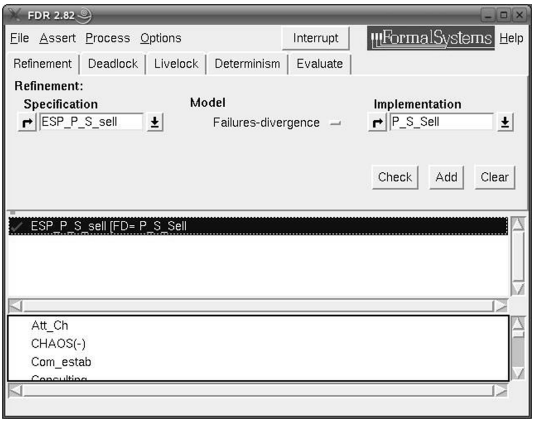


Fig. 6. *Product/Service Sell* TM realization verification screen shot

line) the BPTM realization satisfies the BPTM abstract behaviour as defined according to $ESP_P_S_sell$ failure/divergences semantics formal specification.

4.4 Discussion of Results

According to the result of the BPTM verification shown in Figure 6, we can affirm that the interaction of the *Attention channel* and *Logistic* agents in the realization of the tasks specified by the *Product/Service Sell* BPTM does not provoke deterioration of the QoS required by *Product/Service Sell* BP. Furthermore, the tasks assigned to *Attention channel* and *Logistic* agents are carried out by them within the maximum execution times specified by the respective TM, and *Product/Service Sell* BP will stay waiting for the least possible time only, provided when the *Attention channel* and *Logistic* agents are busy performing the BP. On other hand, these PBPs, could not reach a deadlock state either, i.e., to stay waiting the communication of each other forever. In other words, the verification points out that when a requests of sell a product/service arrive, both *Attention channel* and *Logistic* agents will always satisfy the TM associated to *Product/Service Sell* BP.

5 Conclusions and Future Work

In this paper, we have described how a MC–based, compositional verification framework, called FCVA and the formal design method MEDISTAM–RT for real–time systems can both integrate seamless in the TM software development life–cycle. The support given by CSP+T formal language to our approach allow us to obtain a precise BPTM specification that can be later used for the verification activity, by using state–of–the–art MC tools. In this way, we will help those who are responsible of the BPM activity to verify that an accurately specified TM corresponds to a given BP, i.e., to check whether the BP design fulfills the original QoS specifications in order to satisfy the BP goals and rules. We can therefore affirm that our FCVA integrates, within

the same framework, the activities connected with analysis, design and verification of a TM associated to a critical BP. The application of FCVA to a case study related to CRM business shows the feasibility of our compositional verification vision to verify the behaviour of a BPTM, supported by the formal semantics of the KS.

The future and ongoing work is aimed at the application of FCVA to other BPM case studies; thus, our goal is to achieve the verification of practical cases supported with other automatic verification tools, and to conduct an in–depth research on the verification of BP specifications.

Acknowledgements. We would like to thank A. Grimán, M. Pérez, and X. Vargas for the case study information used in this work. This research was partially supported by National Fund of Science, Technology and Innovation, Venezuela, under contract G–2005000165.

References

1. OMG: Business Process Modeling Notation – version 1.1. Object Management Group, Mass., USA (2008)
2. Eshuis, H.: Semantics and Verification of UML Activity Diagrams for Workflow Modelling. Phd thesis, University of Twente, Enschede, The Netherlands (2002)
3. Duursma, C., Olle, U.: Task model definition and task analysis process. Technical report, Vrije University, Brussels KADSII /M5/VUB/RR/004/2.0 (1994)
4. Paternò, F.: Task Models in Interactive Software Systems. In: Handbook of Software Engineering And Knowledge Engineering: Recent Advances, World Scientific Publishing Co., Inc., River Edge (2001)
5. OASIS: Web Services Business Process Execution Language Version 2.0. OASIS Open, Billerica, USA (2007)
6. Benghazi, K., Capel, M.I., Holgado, J.A., Mendoza, L.E.: A methodological approach to the formal specification of real–time systems by transformation of UML–RT design models. Science of Computer Programming 65(1), 41–56 (2007)
7. Kruchten, P.: The Rational Unified Process: An Introduction, 3rd edn. Addison-Wesley Longman Publishing Co., Inc, Boston (2003)
8. Aalst, W., Hofstede, A., Weske, M.: Business Process Management: A Survey. In: van der Aalst, W.M.P., ter Hofstede, A.H.M., Weske, M. (eds.) BPM 2003. LNCS, vol. 2678, pp. 1–12. Springer, Heidelberg (2003)
9. Schneider, S.: Concurrent and Real-time Systems - The CSP Approach. John Wiley & Sons, Ltd., Chichester (2000)
10. Žic, J.: Time–constrained buffer specifications in CSP+T and Timed CSP. ACM Transaction on Programming Languages and Systems 16(6), 1661–1674 (1994)
11. Formal Systems (Europe) Ltd: Failures–Divergence Refinement – FDR2 User Manual. Formal Systems (Europe) Ltd., Oxford (2005)
12. Eriksson, H.E., Penker, M.: Business Modeling With UML: Business Patterns at Work. John Wiley & Sons, Inc., New York (1998)
13. Clarke, E., Grumberg, O., Peled, D.: Model Checking. The MIT Press, Cambridge (2000)
14. Baier, C., Katoen, J.P.: Principles of Model Checking. The MIT Press, Cambridge (2008)
15. Mendoza, L.E., Capel, M.I., Benghazi, K.: Checking behavioural consistency of UML–RT models through trace–based semantics. In: Proc. 9th International Conference on Enterprise Information Systems (ICEIS 2007), vol. 3, pp. 205–211 (2007)

16. Mendoza, L.E., Capel, M.I.: Consistency checking of UML composite structure diagrams based on trace semantics. In: Meyer, B., Nawrocki, J.R., Walter, B. (eds.) CEE-SET 2007. LNCS, vol. 5082. Springer, Heidelberg (2008)
17. Sifakis, J.: Modeling real-time systems – challenges and work directions. In: Henzinger, T.A., Kirsch, C.M. (eds.) EMSOFT 2001. LNCS, vol. 2211, pp. 373–389. Springer, Heidelberg (2001)
18. Mendoza, L., Marius, A., Pérez, M., Grimán, A.: Critical success factors for a customer relationship management strategy. In: Inf. Softw. Technol. 49(8) (2007)

Actor Relationship Analysis for the i* Framework

Shuichiro Yamamoto[1], Komon Ibe[1], June Verner[2], Karl Cox[2], and Steven Bleistein[2]

[1] Institute of System Science, NTT DATA, Tokyo, Japan
[2] UNSW and Enterprise Analysts Pty., Ltd, Australia
{yamamotosui,komomib}@nttdata.co.jp, june.verner@gmail.com
{karl,steve}@enterpriseanalysts.com.au

Abstract. The i* framework is a goal-oriented approach that addresses organizational IT requirements engineering concerns, and is considered an effective technique for analyzing dependencies between actors. However, the effectiveness and limitations of i* are unclear. When we modelled an industrial case with a large number of actors using i*, we discovered difficulties in (1) validating the completeness of the model, and (2) managing change. To solve these problems, we propose an actor relationship matrix analysis method (ARM) as a precursor to i* modeling, which we found aided in addressing the above two issues. This paper defines our method and demonstrates it with a case study. ARM enables requirements engineers to better ensure completeness of requirements in a repeatable and systematic manner that does not currently exist in the i* framework.

Keywords: Goal oriented requirements engineering, i* framework, Actor relationship analysis.

1 Introduction

In the requirements engineering literature, there is recognition that higher-level organizational concerns are important as they have an impact on IT requirements. i* is an approach that address organizational IT requirements engineering concerns [1]. The i* framework takes an agent-oriented, organizational modeling approach to information systems requirements. i* integrates organizational actors and roles within a goal model representing intentions as a means of modeling requirements for information systems (http://www.cs.toronto.edu/km/istar/, Yamamoto et al 2006). The primary feature of the i* notation is the modeling of intentions of multiple actors within an organizational context [1]. This modeling consists of relationships describing dependencies upon other actors and IT resources in order to achieve organizational goals [1]. The actor dependency relationships include goal, softgoal, resource and task.

However i* is not always effective when used to represent real world industrial problems. Difficulties encountered by practitioners attempting to apply i* in industrial projects is documented in an empirical study in which refinement, complexity management, traceability, and scalability were all found to be either not supported or not well supported by i* [2, 3]. Similarly, difficulties encountered by engineers in applying i* are documented in another industry case study, which concludes that

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 491–500, 2009.
© Springer-Verlag Berlin Heidelberg 2009

"further method engineering work is needed to support the development of scalable i* models" [4]. In particular, "constructing a single model specifying all actors and their dependencies is very difficult due to the number and nature of these dependencies" [4].

Motivated in part by problems in the industrial application of i*, B-SCP is a modeling framework developed expressly for industrial application to large, strategic enterprise IT projects that integrates i* with Jackson Problem Diagrams in order to address issues of scalability and complexity [5, 6].

We found similar issues with the i* framework when we evaluated i* in an industrial case study [7, 8], particularly:

(1) If the number of actors is large, it is difficult to recognize the completeness of actor dependencies with the strategic dependency model of the i* framework due to complexity of the actor relationships.
(2) Because of the complexity of actor dependencies situations surrounding actors can be problematic if changes are required. As the i* framework only describes the dependency relationships among actors for a specific situation if we wish to analyze the effects of change, it is necessary to integrate actor relationships with modelling situations, problems, intentions and actors.

To address the above issues, in this paper we propose an actor relationship matrix analysis method (ARM) as a precursor to i* modeling, which we found aids in addressing the two issues noted above. We demonstrate ARM with a small case study in Section 2. Then in Section 3 we describe a systematic transformation method to develop an i* strategic dependency model based on an ARM. Section 4 provides an industrial case study of the method. Section 5 provides a discussion that considers potential applications of ARM in practice, its usefulness, and integration with strategic dependencies. Section 6 presents our conclusions.

2 ARM

We begin this section with an overview of an ARM including an actor situation matrix (ASM), ARM*, and a strategic dependency model (SD). This is followed in section 2.2 by the definition of an ASM, and in 2.3 by a definition of the various components of an ARM and ARM*. This section includes an example of the development of an ARM and its translation to a SD model.

2.1 Overview

Building an i* development framework consists of the following steps, shown in Figure 1.

(1) Develop an actor situation matrix (ASM) which extracts actors for each situation.

(2) Develop an ARM which defines inner goals for each actor, as well as softgoals, resources, and tasks between each pair of actors.

(3) Develop ARM* integrating ARM for each situation, so that all situations are covered.

(4) Develop strategic dependency (SD) model based on ARM*.

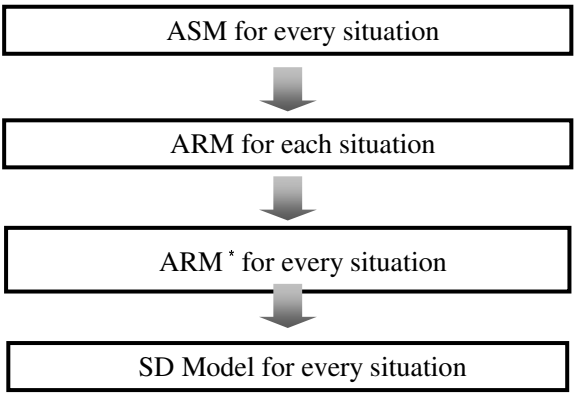


Fig. 1. Stepwise SD model development method

2.2 Actor Situation Matrix: ASM

[Definition 1] - ASM

For situations S1,.., Sm, and actors A1,.., An, the 2 dimensional matrix ASM is defined as follows.

ASM [i,j] = 1 if the situation Si includes actor Aj otherwise 0 ,where 0<i<m and 0<j<n

An ASM defines the relationship between situations and actors and enables an analysis of which actors are associated with each situation. It also enables us to represent all situations in which a specific actor appears. Figure 2 provides an example of an ASM for situations S1 and S2, and actors A, B, and C. Actor C appears in situation S1, but does not appear in situation S2.

Situation	Actor A	Actor B	Actor C
S1	1	0	1
S2	1	1	0

Fig. 2. Actor situation matrix

2.3 Actor Relationship Matrix - ARM

[Definition 2] ARM (A1,· · · ,An)

For a situation S and actors A1,..,An, the 2 dimension matrix ARM (A1,· · · , An) is defined as follows. Here, ASM[S,I] = 1 (0<i<n+1) is assumed.

The 'A' column and rows correspond with actors, A1,.., An. The element for the i-th row and j-th column represents the intention Ik of actor Ai to actor Aj. The intention includes softgoals, goals, resources, and tasks.

The element for the i-th row and i-th column represents the goals of actor Ai.

ARM (A1,・・・, An) [i,i] = {Gk | Gk is a goal of Ai, 0<k<li}

ARM (A1,・・・, An) [i,j] = {Ik | Ik is an intention of Ai to Aj, 0<k< l(i,j), i ≠j}

An ARM is developed for each situation.

dependee / depender	Actor A	Actor B
Actor A	$\{(G_{AA}, S_1), (G_{AA}, S_2)\}$	$\{(I_{AB}, S_1), (I_{AB}, S_2)\}$
Actor B	$\{(I_{BA}, S_1), (I_{BA}, S_2)\}$	$\{(G_{BB}, S_1), (G_{BB}, S_2)\}$

Fig. 3. Actor relationship matrix

[Definition 3] Element set of ARM

S (ARM (A1,・・・, An)) = (∗0<i<n+1 ARM (A1,・・・, An) [i,i])

∗(∗0<i<n+1, 0<j<n+1, i ≠ j ARM (A1,・・・,An) [i,j])

Figure 3 shows an ARM for actors A, B, and C.

ARM (A, B, C) [A,B] describes intentions from the depender A to its dependee B.

The ARM (A, B, C) [A, A] shows the inner goals of actor A.

Although the example provided above describes a three-dimensional ARM, it is easy to define an *n* dimensional ARM that represents the dependency relationships among n actors and their goals.

dependee / depender	Actor A	Actor B	Actor C
Actor A	G_{AA}	I_{AB}	I_{AC}
Actor B	I_{BA}	G_{BB}	I_{BC}
Actor C	I_{CA}	I_{CB}	G_{CC}

Fig. 4. Integrated actor relationship matrix

2.4 Total Actor Relationship Matrix: ARM*

[Definition 4] ARM*

For actors A1,.., An and situations S1,.., Sk, the total actor relationship matrix, ARM*, is defined as follows:

ARM*(A1,.., An)[i,j] = { (Iij,Sk) | actor Ai and Aj appear in situation Sk, ASM[k,i] = ASM[k,j]=1, i≠j }

ARM*(A1,.., An)[i,i] = { (Gii,Sk) |ASM(k,i) =1 } }

For each actor, related situations are enumerated by ASM. Then the dependent intentions are elicited by using ARM* for each actor. By combining ASM and ARM*, the intentions of actors are easily gathered and the coverage improved. Figure 4 shows an example of ARM*.

[Method 1] Transformation from ARM* to a SD model of an i*framework
The following are transformation rules for generating an SD model from ARM*.

(Rule 1) Elements of ARM*(A_1,.., A_n) [i,j] is transformed to the dependum form Ai to Aj.
(Rule 2) The dependum is connected to depender and dependee actors by arrows as follows.
Ai→-- dependum →-- Aj
(Rule 3) For each dependum, situation Sx is annotated. The annotation is generated by the fact that in ARM*(A_1,.., A_n)[i,j] contains an element (Iij,Sx).

Figure 5 shows a portion of generated SD model from ARM using Rule 3.

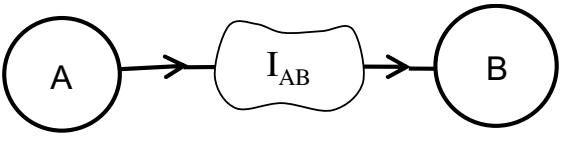


Fig. 5. SD model based on ARM element I_{AB}

3 Example

We now illustrate the proposed method using situations and actors in a CD player with the ASM, progressing through the ARM* to an SD model.

3.1 ASM

Four situations are assumed for the CD play cycle.

S1: User inserts CD to the CD player.
S2: User pushes the play button to listen to the CD.
S3: User pushes the halt button to stop the CD.
S4: User ejects the CD by pressing the eject button.

Actors are identified for the ASM for each situation. Actors in S1 are CD reader, CD control, and User. Actors in S2 are CD reader, CD control, and Motor. The ASM developed is illustrated in Figure 6.

Situation	CD sensor	CD control	User	Motor
S1:Insert CD	1	1	1	0
S2:Play CD	1	1	1	1
S3:Halt CD	0	1	1	1
S4:Eject CD	1	1	1	0

Fig. 6. Example of ASM

3.2 ARM*

First, we develop an ARM for each situation. Then we integrate the four ARMs into one single ARM*. Situation S1 and S3 have the same set of actors. S1 and S2 have different sets of actors as well as a different set of softgoals.

In S1, the CD reader senses the CD from User and notifies the existence of CD to CD control. In S2, user wants to play CD by pushing the play button. Then CD control lets the motor rotate. The ARM* example for the CD player problem is shown in Figure 7.

3.3 SD Model

Using the transformation rules from Section 2 we can now develop SD models. The integrated SD model based on ARM* is shown in Figure 8.

Here, we can describe one actor in an SD model that has multiple dependency relationships extracted from multiple situations. This also shows that an SD model contains common actors and softgoals in different situations. The actors and softgoals also are annotated according to situations, Sx. This information is useful as the evidence showing the necessity of these actors and softgoals.

4 Case Study

ARM was applied to the requirements review for a Japanese financial system specification [7]. The specification included 14 actors and 152 softgoals. The 3 developers reviewed the specification using ARM over 2 hours. The number of actors was the same before and after the review. However, the number of softgoals increased by 25% after the review. Figure 9 shows the result of the case study. Although the number of actors was the same, both numbers of self and expected softgoals were increased. This means that ARM is effective in eliciting softgoals that were omitted in the requirements specification.

Actually, the developers of the IT division of the financial company discussed with us the effectiveness of ARM as follows:

	CD reader	CD control	User	Motor
CD reader	{(Sense CD,S1)}	{(Notify Eject,S4)}	{(Insert CD,S1)}	
CD control	{(Notify Insert,S1)}	{(Play,S2),(Halt,S3),(Eject,S4)}	{(Play button ON,S2)} {(Halt button ON, S3)} {(Eject button ON, S4)}	{(Notify rotate,S2),(Halt,S3)}
User		{(Play,S2),(Halt,S3),(Eject,S4)}	{(Listen CD, S2)}	
Motor				{(Rotate,S2),(Halt,S3)}

Fig 7. Example of ARM* for a CD player

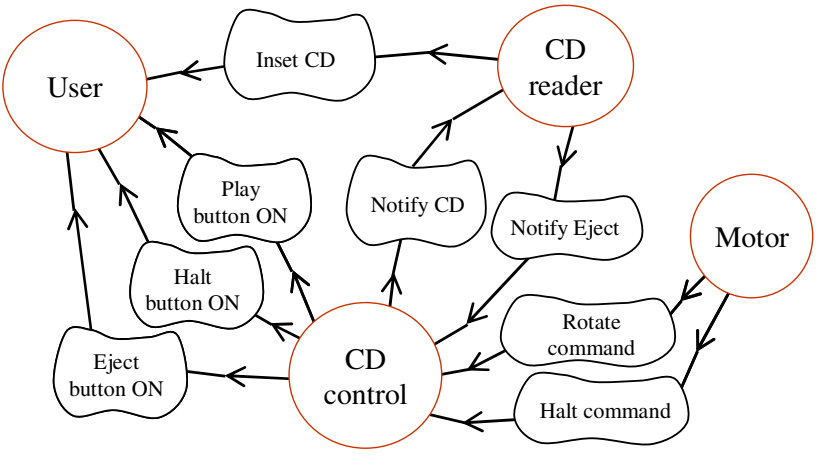


Fig. 8. Example of SD model for CD player

	Before review	After review	Difference
Actor	14	14	0
Self softgoal	17	30	13
Expected softgoal	135	160	25
softgoal in total	152	190	38

Fig. 9. Result of ARM based review

ARM is useful in checking for incompleteness in requirements specifications. It is always difficult to understand the existence of necessary requirements that are not written into the specifications. ARM provided an easy and practical means to elicit the unwritten requirements by using matrices.

5 Discussion

We further discuss the ARM including its usefulness and the equivalence and integration of ARM with SDs.

5.1 Usefulness of ARM

ARM is useful in confirming the coverage of dependency relationships among actors thus helping ensure completeness of a SD model. In contrast, when using only a SD model in isolation, which is typical in i* framework modeling, it is difficult to analyze the coverage of dependencies among actors, thus increasing the risk of an incomplete SD model. From our experience of teaching the i* framework, many students reported difficulties in understanding and confirming the coverage an i* SD model.

An ARM is also useful if we need to find commonality of actors. If two actors have common ARM rows or columns, these actors may be equivalent. If two actors have the same subset of ARM rows or columns, there may be a common subset of actors. This kind of commonality relationship analysis is difficult when using an i* framework in isolation.

If the ARM is a sparse matrix we have an actor that has few dependency relationships with other actors. These kinds of actors are very easy to find in an ARM.

5.2 Equivalence of SD Models Using ARM

It is not easy to decide if two complex SD models are equivalent or not. However, it is easy to compare the equivalence of the corresponding ARM for two SD models.

If we have two SD models, M1 and M2 we can define a map F from the SD model to an ARM.

Now we have two ARMs, F(M1) and F(M2).

If F (M1) = F(M2) then there are actors Ax and Ay for any element of F(M1) =F(M2).

Now we assume that some actors Ax and Ay are in M1 but not in M2. By the definition of F, we could have an element Ixy of F(M1)[x,y]. This element Ixy will also be an element of F(M2)[x,y]. This shows actors Ax and Ay should also be included in M2. This is a contradiction. Therefore, the actors of M1 and M2 are equivalent.

Then we assume an element Ixy is in M1 but not in M2. In this case we can derive a contradiction by the definition of F.

According to the above discussion for M1 and M2, M1=M2 if and only if F(M1)=F(M2).

This could be applied to tracking of requirements as they evolve or change over time, as it is possible to identify incremental changes between different versions of an SD model.

5.3 Integration of SD Models with ARM

It is not an easy task to integrate multiple SD models. It is also difficult to extract common portions from different SD models. However, this kind of management task is essential if we wish to reuse i* framework requirements models.

A proposed mapping, F, is very powerful for the management of this kind of SD model integration.

Now we consider the integration of two SD models M1, M2. We can have two ARMs - F(M1) and F(M2).

Next we develop a new mapping F_{12} (M1,M2) that can be defined as F(M1) + F(M2). Here a suffix set of F(M) is called Suf(F(M)). The suffix set corresponds to the actors of M.

(1) If i,j is in Suf(F(M1)) but not in Suf(F(M2)) then F_{12}(M1,M2)[i,j] =F(M1)[i,j]

(2) If i,j is in Suf(F(M2)) but not in Suf(F(M1)) then F_{12}(M1,M2)[i,j] =F(M2)[i,j]

(3) If i,j is both in Suf(F(M1)) and Suf(F(M2)) and F(M1) [i,j]= F(M2) [i,j] then F_{12}(M1,M2)[i,j] =F(M1)[i,j]

(4) If i,j is both in Suf(F(M1)) and Suf(F(M2)) and F(M1) [i,j]≠F(M2) [i,j] then F_{12}(M1,M2)[i,j] =F(M1)[i,j]*F(M2)[i,j]

(5) If (i is in Suf(F(M1)) and not in Suf(F(M2)) and j is not in Suf(F(M1)) and in Suf(F(M2))) or (i is not in Suf(F(M1)) and in Suf(F(M2)) and j is in Suf(F(M1)) and not in Suf(F(M2))) then F_{12}(M1,M2)[i,j] is empty.

The SD model transformed from F_{12}(M1,M2) will become the integrated SD model for M1 and M2. This has potential application in large systems integration projects, such as the merger of information systems for two banks, for example.

5.4 Correspondence between Situations and Inner Goals

As the inner goals of an actor change with different situations, the correspondence between situations and an actor needs to be managed. It is hence necessary to extend the ARM method to treat inner goal decomposition for different situations. Inner goals may be decomposed for the different situations in a Strategic Rational model.

6 Conclusions

In this paper, we propose a SD model development method integrating actor dependency matrixes for situations.

The ARM method has potential for the following:

1. A systemic means of identifying relevant actors and SDs prior to construction an i* SD model
2. Ensuring completeness of coverage
3. Tracking evolution of requirements
4. Identifying reusable parts
5. Integration of SD models in a systematic manner
6. Identifying redundancy or unnecessary model detail

ARM can potentially be useful as an analytical precursor to i* modeling, to help ensure completeness of requirements. We also believe, as indicated in Section 4, that ARM might be applied to large systems integration projects, rationalization of requirements, and management of evolving and changing requirements.

Future research includes extending our work into the development of SR models and full experimental evaluation in industry in order to quantify the effectiveness of the method in a number of different situations.

References

1. Yu, E.: Towards Modeling and Reasoning Support for Early-Phase Requirements Engineering. In: 3rd IEEE International Symposium on Requirements Engineering (RE 1997), pp. 226–235 (1997)
2. Pastor, O., Martínez, A., Estrada, H.: Some lessons learned from using i* modeling in practice. In: International RESG Workshop on Organizational Modeling, City University, London (2005)
3. Estrada, H., Martínez, A., Pastor, O., Mylopoulos, J.: An experimental 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)
4. Maiden, N., Jones, S., Manning, S., Greenwood, J., Renou, L.: Model-driven requirements engineering: Synchronising models in an air traffic management case study. In: 16th International Conference on Advanced Information Systems Engineering, pp. 368–383 (2004), http://www.cs.toronto.edu/km/istar/ihomepage (retrieved, December 11, 2008)
5. Bleistein, S., Cox, K., Verner, J.: Validating strategic alignment of organizational IT requirements using goal modeling and problem diagrams. Journal of Systems and Software 79, 362–378 (2006)
6. Bleistein, D., Cox, K., Verner, J., Phalp, K.: B-SCP: A requirements analysis framework for validating strategic alignment of organizational IT based on strategy, context, and process. Information and Software Technology 46, 846–868 (2006)
7. Ibe, K., Saito, S., Yamamoto, S.: ARM: Actor Relationship Matrix. In: Joint Conference on Knowledge Based Software Engineering 2008, pp. 423–426. IOS Press, Amsterdam (2008)
8. Yamamoto, S., Kaiya, H., Cox, K., Bleistein, S.: Goal Oriented Requirements Engineering: Trends and Issues. IEICE Transactions on Information and Systems E89-D(11), 2701–2711 (2006)

Towards Self-healing Execution of Business Processes Based on Rules

Mohamed Boukhebouze[1], Youssef Amghar[1],
Aïcha-Nabila Benharkat[1], and Zakaria Maamar[2]

[1] Université de Lyon, CNRS, INSA-Lyon, LIRIS, UMR5205, F-69621, France
{mohamed.boukhebouze,youssef.amghar,
nabila.benharkat}@insa-lyon.fr
[2] CIT, Zayed University, Dubai, U.A.E.
zakaria.maamar@zu.ac.ae

Abstract. In this paper we discuss the need to offer a self-healing execution of a business process within the BP-FAMA framework (Business Process Framework for Agility of Modelling and Analysis) presented in [1]. This will be done by identifying errors in the process specification and reacting to possible performance failures in order to drive the process execution towards a stable situation. To achieve our objective, we propose to model the high-level process by using a new declarative language based on business rules called BbBPDL (Rules based Business Process Description Language). In this language, a business rule has an Event-Condition-Action-Post condition-Post event-Compensation (ECA2PC) format. This allows translating a process into a cause/effect graph that is analyzed for the sake of ensuring the reliably of the business processes.

Keywords: Business processes modeling; business rules; declarative language and self-healing of business process.

1 Introduction

The process modeling is then an important step in the business process management, because it allows specifying the business knowledge of a company. For this reason, it must be based on powerful languages in order to give a full business process description. In this context two standards are proposed: a common graphical notation for modeling tools called BPMN, and process execution language called BPEL, to make processes portable on different platforms. These two specifications are now stable and adapted to business needs. Several editors have adopted and included them in their tools.

However, these specifications have to be checked before their implementation, unfortunately, these standards focus more on the business description level, including functional aspects of a process, without providing mechanisms to support the specifications verification. Indeed, both the reliability of the process and maintenance costs drives us to give a great attention to verification issues.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 501–512, 2009.
© Springer-Verlag Berlin Heidelberg 2009

For this raison, we have proposed in [1] a new framework called **BP-FAMA** that stands for **B**usiness **P**rocess **F**ramework for **A**gility of **M**odeling and **A**nalysis. BP-FAMA development is in progress its objective is to improve the management of business processes in terms of flexibility and verification. Indeed, we consider reviewing the way business processes are managed. For instance we consider offering a flexible way to model processes so that changes in regulations are handled through some self-healing mechanisms. These changes may raise exceptions at run-time if not properly reflected on these processes. For this reason, in this paper we focus to how to self-heal execution of a business process?

To achieve this objective we propose a new declarative language for describing business processes. This language is called **RbBPDL** (**R**ule-**b**ased **B**usiness **P**rocess **D**escription **L**anguage). The objective of RbBPDL is to define, in a declarative way, the high level business processes using a set of business rules. RbBPDL process logic can be summarized with a set of business rules. The sequences of these rules define the behavior of a process. Indeed, RbBPDL uses business rules where each business rule implements the ECA2PC format (Event-Condition-Action-Post condition-Post event -Compensation). The great advantage of ECA2PC is that RbBPDL processes can be easily translated into a cause/effect graph. Analyzing this graph guarantees the self-healing execution of a business process by identifying, in the modeling phase, any risk of exceptions (verification step) and managing these exceptions, in the execution phase, in order to ensure a proper functioning of a process (exceptions handling step).

The rest of this paper is organized as follows. We introduce in section 2 the motivations for our work. In section 3 we describe the BP-FAMA architecture. Section 4 specifies in more detail the new language RbBPDL. In section 5, we explain how RbBPDL process can be self-healed. We wrap up the paper with a conclusion and some directions for future works.

2 Related Work

The main motivations of the BP-FAMA framework stem out of the importance of improving business process management in terms of flexibility and verification. But, these characteristics are highly inter-related, which increases the complexity of satisfying them all. Real business processes tend to be less flexible and difficult to analyze due to continuous changes in regulations and policies.

Indeed, the first type of business process modeling languages known as imperative languages, such as BPEL [2] and XPDL [3], focus on how the various activities in a process are ordered for execution purposes. These languages provide a good level of expressivity and several verification techniques have been proposed to ensure the reliability of the processes that are defined using imperative languages (e.g., Petri Net). However, the use of these languages forces designer to describe explicitly the execution scenarios in the modeling phase, which is not very convenient. This makes business processes rigid and difficult to maintain, which is not in line with the dynamic nature of organizations. The explicit definition of how a process should behave in the modeling phase compromises this process flexibility.

A second type of business process modeling languages known as declarative languages has been proposed to deal with the flexibility requirement in a proper way. Examples of these languages include ConDec [4], PENELOPE [5] and SBVR [6] focus on what to do rather than how to do. The process logic in these languages can be summarized using a set of business rules that implement the policies of organizations. Therefore, the execution scenarios are implicitly done in the modeling phase. This could avoid listing all the possible scenarios of execution in the modeling phase, which is difficult to obtain [7] and can result into rigid models. In our framework, we are particularly interested in using a declarative language based on business rules so that the flexibility of these processes can be achieved. However, the changes may raise exceptions at run-time if not properly reflected on these processes.

Indeed, companies must be based on robust business processes to achieve their objective. The process reliability is a crucial issue because they automate all or part of the company value chain and at the same time capitalize on their information system. An erroneous business process can have economically grievous effect. For this reason, self-healing system is used.

By self-healing we mean the ability to detect and isolate the failed component, fix or replace the component, and finally, reintroduce the repaired or replaced component without any apparent application disruption [8]. Indeed, a poorly-formed business process has major negative consequences on the continuity of operations, which result in raising exceptions. Note that, generally the term exception is used to characterize all situations that arise during the execution of a process [9]. Several studies have looked into the nature of exceptions that could affect a business process [9] and [10].

It is always recommended that discovering, diagnosing, and reacting to exceptions should be automatically done so that self-healing systems could be developed, which requires less human assistance. The advantage of such systems is to minimize disruptions so that business process continuity and availability are maintained all times [10]. Self-healing mechanism can be summarized with the following steps: Faults are tracked and then signaled to the designer during process specification using several techniques like design verification and formal (e.g., Petri Nets (PN) [11] and [12]. The combination of both techniques proves profitable to analyze a business process modeling. However, the existing algorithms that implement these techniques are time consuming and complex. In addition, a change in a part of the process results in a complete re-verification of this process. Exceptions are handled trying to react in order to drive the execution of the process towards a stable situation. Indeed, to remedy the effects of an exception, several alternative actions can be launched like: (1) compensate code, (2) substitute the failed Web service (or application), which is responsible for the execution of process activity, with a functionally equivalent Web service [10] or (3) rollback the process model (reversing process engineering).

However, self-healing systems require an execution scenario to ensure proper process functioning. Unfortunately, the various declarative process modeling languages do not allow having an explicit execution scenario. As a result a more powerful way to translate a business process into a formal model and ensure its automatic verification through a realistic scenario is deemed appropriate.

To wrap up this section, there is a need for a new powerful format that would help solve the problems of process modeling rigidity and business process reliably. We propose a new declarative language called RbBPDL that is built on top of the

ECA2PC format. This would allow translating a process into a graph of rules where the nodes represent the rules and the edges represent the relationship between these rules. By doing this, we favor the flexibility of business processes modeling by determining the rules that are connected to each other and the rules that are subject to changes, and favor a self-healing execution of business processes through proper detection and reaction of exceptions.

3 The BP-FAMA Architecture

The objective of the BP-FAMA is to improve the functioning of BPMS (Fig.1). In fact, this platform can:

- Favor the flexibility of the modeling business processes. For this reason, we propose a new declarative modeling language called RbBPDL (Rule-based Business Process Description Language) which is based on XML and consider a process as a set of business rules where each rule is represented using the ECA2PC format. To increase the visibility and understanding of an RbBPDL process, a graphical representation can be adopted. For this reason, the notations of the URML language can be used. Indeed, the latter is proposed by Wagner in [13] in order to describe the rules with graphical notations and meta-models inherited from its ancestor UML. The BP-FAMA editor uses URML to graphical describe a business process and serializes rules using the language RbBPDL.
- Favor the reliability of the process. For this reason, the process parts that can be a potential source of functional errors is identified by analyzing the cause/effect graph associated to the process by lunching an self-healing system, in demon, to manage possible exceptions and try to react in order to drive the execution of the process towards a stable situation. These exceptions are managed by offering a substitute mechanism to replace the failed services (or application) as we will show thereafter. In addition, the roll-backing is allowed by returning to the rules in question in process model in the case of detecting an exception in the execution phase.

4 The RbBPDL Language

RbBPDL process logic can be summarized with a set of business rules. The sequences of these rules define the behavior of a process. Indeed, the rule sequences represent implicitly the control-flow of elements that must be performed in a process. However, to provide a flexible process modeling we need to connect rules together. The verification process requires the implementation of a scenario to test the proper functioning of the process. Business rules in RbBPDL must be based on a format that allows reaching our objective. For this reason, we propose the format ECA2PC which is defined as:

ON ***<Event>***
IF ***<Condition>***
DO ***<Action>***
Check ***<Post condition>***
Trigger ***<Post event>***
On exception **<Compensation>**

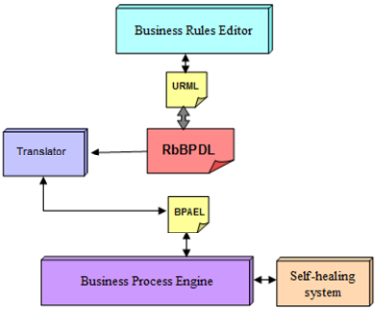


Fig. 1. The BP-FAMA architecture

The semantics attached to an ECA2PC rule is: when an event occurs, the condition is verified, if this latter is satisfied, then the party action is executed by taking into account the rule attributes. The execution of the action causes the post-event party. The rule is validated if the post condition is satisfied. Finally, if an exception occurs, the compensation code is lunched in order to remedy the effects of this exception. Note that, the compensation part is deactivated by default and it will be automatically activated by the self-healing system as we will show thereafter.

ECA2PC format allows an implicit description of sequences of rules. Each rule may trigger one or more rules. But, the originality of this format is the fact that the post events are explicitly described. As a result, the rule sequence can be automatically deducted. Indeed, according to a post event of a rule, we can detect which rules will be triggered.

To represent this format as well as the various elements in a business process, the RbBPDL language is XML-based and inspired by BPEL and XPDL. Fig.2 represents the XML syntax of this new language RbBPDL. Indeed, the participants, variables, business activities, and events are represented with XML tags like "Participants", "Variables", etc. Additional details could be added to the representation so that a complete definition of the process elements is offered. These details could concern type and role of each participant, data types of each variable, input/output parameters of each activity… etc. Note that, there are two categories of events: (1) Simple events describe the occurrence of a predefined situation in the system such as: activity events (start, end, cancellation, error), process events (error trigger), time events (timer), and external events (reception message signal). (2) Complex events that combine simple and/or composed events using constructors such as disjunction, conjunction … etc.

Another point is that, as business rules are described using ECA2PC, they are represented as follows:

- **OnEvent:** all events that activate a rule.
- **Precondition:** predicates upon which the execution of an action depends.
- **Action:** set of instructions to be executed if a precondition is true. For this we can use predefined instructions such as:
  - *Copy*: to copy data from one place to another
  - *Discover*: to find a service that performs a given activity in a given registry
  - *Execute***:** to execute the activity by a participant
  - *Cancel*: to cancel the execution of the activity

- **Postcondition:** predicates upon which the validation of a rule depends.
- **PostEvent:** the set of events triggered by the execution of all instructions of the rule's action.

```
<process>
  <Participants >
     <Participant/> *
 </Participants >
  < Variables >
     <Variable/> *
  </ Variables >
  < BusinessActivities >
     <BusinessActivitie/> *
  </ BusinessActivities >
  < Events >
     <AtomicEvent/> *
     <CompositeEvent/> *
  </ Events >
  < BusinessRules >
    < BusinessRule >
        <OnEvent/>
        <PreCondition/>
        <Action/>
        <PostCondition/>
        <PostEvent/>
   </ BusinessRule >
  </ BusinessRules >
</process>
```

Fig. 2. The global structure of the RbBPDL language

```
<!---------- R1 ---------->

<Business_Rule Name= "R1" Version="1.0">
  <OnEvents> $Recieve_Order </OnEvents>
  <PreConditions> $Info_Costumer != "" </PreConditions>
  <Action>
    <Execute>
      <Operation>$Costumer_Verification"</Operation>
      <InputVariableName>$Info_Costumer</InputVariableName>
      <OutputVariableName>$CostumerRegistred </InputVariableName>
      <Performer>$Commercial_Service</Performer>
    </Execute>
  </Action>
  <PostConditions>  </PostConditions>
  <PostEvents> $End_Costumer_Verification </PostEvents>
</Business_Rule>
```

Fig. 3. One rule of the RbBPDL purchase order process

4.1 Use Case

In this section we introduce the famous example of purchase order process to illustrate the RbBPDL language. Upon order receipt from a costumer (we assume a returning one), then the process simultaneously calculates the initial price of the order and selects a shipper. When both tasks are completed, the final price is calculated and a purchase order is sent to the costumer.

In our new declarative language, a process is seen as a set of decisions and policies. These decisions and policies are defined by a set of business rules. For example Rule R_1 expresses the policy of receiving an order (fig.3). Indeed, during an occurrence the event (<OnEvent> $ Receive_Order </ OnEvent>) the rule is triggered and ensures that the information is valid (<PreCondition> $ info_Costumer! = "" < / Precondition>). If the information is valid, then the action's instruction <Execute> will be executed. This instruction specifies that a given business activity must be performed (<Operation> $Costumer_Verification </ Operation> in our example) by specifying the input/output parameters (<InputVariableName> $ info_Costumer </ InputVariableName> and <OutputVariableName> $CostumerRegistration </ OutputVariableName>). In the order to increase the visibility and understanding of the process representation, a URML graphical representation is given in figure 4. Indeed, the control flow of the RbBPDL activities process is described in an implicitly way based on the ECA2PC format.

In our example, the post event of rule R_1 allows trigging at the same time rule R_2 (defines the policy of calculating an initial price) and rule R_3 (defines the policy of shipping). In this way, the business activities of these two rules will be executed in parallel. In the same way we can define implicitly dependencies between activities, as sequential routing (a post event of a rule triggers another) or synchronization (after all events of two rules triggers a rule).

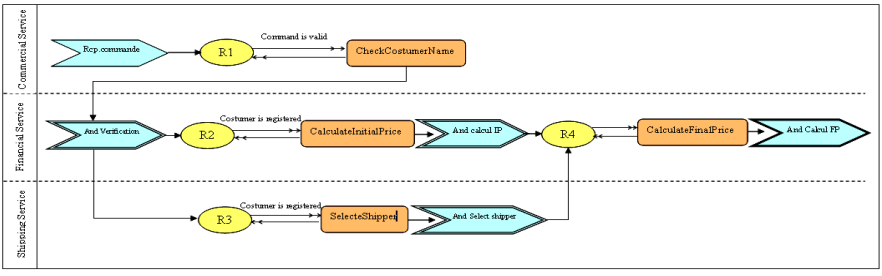


Fig. 4. A graphical representation of the RbBPDL purchase order process

However, in a dynamic environment, companies need to be flexible business so they can quickly react to changes of several parts of a process. But, a change in a process element may raise exceptions at run-time if not properly reflected on these processes. For this reason, BP-FAMA manages these changes through some self-healing mechanisms in order to keep the consistence of the business process.

5 Self-healing Execution of Business Processes

The aim of our framework is to ensure the reliability of the business process by self-healing the execution. To this end, we propose to follow a self-healing strategy for the process RbBPDL on the basis of the ECA2PC format. This strategy requires passing through two major steps: (1) **Exceptions recognition**: the process is verified, in the modeling phase, to identify potential risks of exceptions. (2) **Exceptions handling:** an exception handling is launched in parallel with the execution of the process to intercept the exceptions when it take place and react in order to drive the process execution towards a stable situation.

5.1 Exception Recognition

Exception recognition attempts to identify any risk of exception before the implementation of a process. In this paper we are interested in detecting exceptions that are related to functional coherence of a business process. Such exceptions could come from a poor design, for example: infinite loops and process non-termination. Despite the use of RbBPDL, designers are not immune from making functional mistakes such as live-lock, dead-lock or dead activities. To help designers in detecting early these errors, it is useful to perform a high-level modeling verification in order to provide a reliable operational process. However, to identify these functional errors we should have a process data state. Moreover, this verification cannot be done if an execution scenario is not available. In the case of a declarative modeling it is often difficult to have such a scenario at the modeling time.

To address these problems, we propose to translate the business process into a **cause/effect graph** (Fig.5.A). The vertices of this graph represent the business rules of the business process, and the arcs represent the cause/effect relationship between the various rules. Cause/Effect relationship relates a rule (cause rule) that activates another rule execution (effect rule). As a result, the execution of a cause rule's action

triggers a post event, which necessary activates the effect rule. Thanks to this relationship, the order of RbBPDL process activities can be defined by describing the post events based on ECA2PC. In our previous example, the performance of the R_1's action (costumer verification) will trigger the post event end costumer verification. This latter is the event activator of rule R_2. Due to this, there is a cause and effect relationship between R_1 and R_2 (fig.5.A). In this way, a cause/effect graph is formally defined as follows:

Definition 1
Let be:

- E: a set of the process events
- E_{ri} : a set of the events which active the rule R_i such as: $E_{ri} \subseteq E$
- PsE_{ri} : a set of events which are trigged after the execution of the action of the rule R_i such as: $PsE_{ri} \subseteq E$

A cause/effect graph is a directed graph G_r (R, Y) with

- R is a set of the vertices which represent business rules
- Y is a set of the arcs which represent cause/Effect relationship between rules such as: the rule R_i causes the execution of the rule R_j if $E_{rj} \subseteq PsE_{ri}$

The use of the cause/effect graph for verification of an RbBPDL process is backed by the fact that this latter represents how the process rules set is activated. As a result, a cause/effect graph formalizes the RbBPDL process functioning.

For illustration purposes we adopt the live-lock case. This case occurs if a sub set of rules behave like an infinite loop, which puts a process in an endless state. This could be due to a poor analysis of the rules that are executed. In the previous example, if rule R_5 is changed to allow customers add articles to the same bill and rule R_6 is added later which allows saving the bill (figure 5.B), then the new rule R_5 will rerun the process by activating rule R_1. As a result, the cause/effect graph contains two circuits (R_1, R_2, R_4 and R_5) and (R_1, R_3, R_4 and R_5). Both circuits represent loops in the process and both may be infinite. To determinate whether a circuit in a cause/effect graph can be terminated, we need to have a data state. However, in process modeling, such a data state does not exist. For this reason, each circuit could be now considered as a risk of infinite loop. As a result, rules in each circuit will be identified for testing in the execution phase.

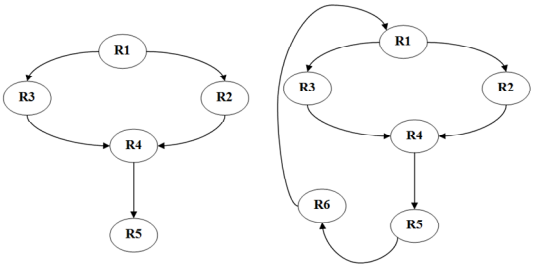


Fig. 5. The cause/effect graph of the purchase order process

5.2 Exceptions Handling

As mentioned in the previous section, exception recognition attempts to detect risks of exceptions by identifying the process part that can possibly cause such exceptions. However, an exception handling step is necessary to monitor these parties at run-time, and to act actions in case these exceptions become effective. The aim of this exception handling step is to avoid the business process to be in an unstable situation. For this reason, the exception handling is launched in parallel with the execution of the process. In this way, this exception handling tries to respond to a situation that would destabilize the process performance by executing compensation codes. To do this, the exception recognition step injects automatically an exceptions handling code into the rules which will likely lead to exceptions. The aim of this code is to verify whether an exception occurred in the executable process. In case the exception occurs, this code launches an alternative remedy of the exceptional effect by activating the compensation part of the rules in question and launching several alternative actions:

(1) Compensation Code. Aims at compensating the exception effects by executing the code necessary to drive the process execution towards a stable situation. In BP-FAMA framework, the compensation codes are implemented as web services called compensate service. Indeed, each compensation service is specified to one exception. For this raison, in the Execute instruction is used to execute the specific compensation service that corresponds to a specific exception (fig. 6). When translating RbBPDL process into an execution process code (as BPEL), the Execute instruction will correspond to a simple web service invocation (this will correspond to the invoke activity in BPEL language).

(2) Substitution. Aims at replacing a failed Web service (or application) with a functionally equivalent Web service (or application) [11]. This mechanism can be done manually by specifying, in the compensation part, a Web service that replaces the current failure web service. The substitution can also be done automatically by discovering discovers web services among several candidates which is most suitable to replace the current failure web service. To do this we use the Discover instruction to find a given Web service in a given registry (fig.7). For this reason, before translating RbBPDL process into an execution process code (as BPEL), service discovery module of the BP-FAMA framework selects one suitable Web service. In this way, this letter will be invoked in the executable business process (by using invoke activity in BPEL language).

(3) Roll-backing. Aims at allowing roll-backing to the rules in question in process model in the case of detecting an exception in the execution phase. To do this, business rules are tracked and identified in a the operational business process by translating the RbBPDL language into the BPAEL language (Business Process Agile Execution Language) proposed in our previous work [1] and which is based entirely on his ancestor BPEL and extends with location and identification of business rules in an executable process. This is done by adding a new activity structured called "RULE".

In following we detail how the exceptions handling can manage live-lock exception. Indeed, as we saw previously, due to lack of data state in the modeling

phase, the exceptions recognition cannot determine the finite nature of a circuit of a cause/effect graph. To this end, a live-lock exception handling code will be added into the code of the all rules of each circuit of cause/effect graph. However, to optimize the addition of this code, the live-lock exception handling code will be injected into only two specific rules by circuit: the first is added to the code of the starting rule circuit. The second is added to the code of the ending rule circuit. The justification for this choice is explained thereafter.

For instance, to manage the two circuits of the cause/effect graph in the purchase order process (Fig.5.B), a live-lock exception handling code is added per circuit in the rule R_1 (the starting rule of the two circuits) and in the rule R_6 (the ending rule of the two circuits). In this way, the Live-lock exception handling code enables the monitoring of this circuit in the execution phase by checking if the process is in a state where it constantly rotates (live-lock) by bases on the data state of the process. A state data is defined as follows:

Definition 2. A data state of one process at a time t, noted $\beta(t)$, is the vector of process values at a time t.

The live-lock exception handling code will test the data process by considering the following property.

Property. In a cause/effect graph, a circuit is finite if the two following properties are verified:

1) All the process variables belong to a boundary interval

The change of data is respected as $\forall t, \neg\exists t' / \beta(t) = \beta(t')$

According to this property, completing a loop requires that the data state changes over time, i.e., at least one of the process variables must change in each loop iteration. In the previous example, live-lock exception handling will ensure that the data state changes in each iteration (adding an article, deletion of an article, etc.). If the process receives the same information in one command instance this means that the process has entered an infinite loop.

Based on this property, the a live-lock exception handling code (red parties in the fig.6 and fig.7) of a starting rule in a circuit (in the preceding example R_1) compares the data state of the current iteration with the data states of all previous loop iterations. If the code detects a recurring data state, the loop is infinite. In this case, the code will launch an alternative remedy in order to lead the process execution to a valid situation. Indeed, the first remedy can be a compensation code that consist of, in BP-FAMA framework, executing the $Compensate_NoTerminateProcess_Service, as it's indicate in Compensate part of the rule R_1 (fig.6). This service compensates the live-lock effects by for example, stopping the properly process or got out of the loop to continue the execution of the process. In this way, the operational team in charge of deploying the process can define actions to be done if the process turns in circles. When translating RbBPDL process into an execution BPEL process code, this Execute instruction will correspond to invocate the Compensate no terminate process service by using the BPEL invoke activity.

Second alternative remedy can be a service substitution that consist of, in BP-FAMA framework, the specification, in the Discover instruction (fig.7), the UDD

registry and the suitable quality of the Web service candidate that will replace the current failure web service (for instance, the access cost is at most 5€/access). In this way, before translating RbBPDL process into an execution BPEL process code, service discovery module of the BP-FAMA framework (fig.1) will select one suitable Web service. After that, the service selected will be invoked in the executable business process by using BPEL invoke activity.

Note that the substitute is more suitable for the operational exceptions because these exceptions relate to events which are not be modeled by the designer and which are liable to cause exceptions. In fact, these events are infrequent and unexpected for example unavailability of one or more resources when an activity instance in process wants access.

```
<BusinessRules name="R1" version ="1.0" >
        <OnEvents> $Recieve_Order</OnEvents>
        <PreCondition>
                ($Info_Costumer != "")
            and ($ElementExiste(DataStateStack, CurrentDataSate)== false )
        </PreCondition>
        <Action>
          <Execute>
                <Operation> $Costume_Verification </Operation>
                <InputVariableName> $Info_Costumer </InputVariableName>
                <OutVariableName> $CostumerRegistred </OutVariableName>
                <Performer> $Costume_Verification </Performer>
          </Execute>
          <Execute>
                <Operation> $ GetDataState() </Operation>
                <InputVariableName> $CurrentDataSate </InputVariableName>
                <Performer> $Expetion_Handler </Performer>
          </Execute>

        </Action>
        <PostCondition> </PostCondition>
        <PostEvents> $End_Costume_Verification</PostEvents>
        <Compensate>
          <Execute>
                <Operation> $Compensate_NoTerminateProcess </Operation>
                <Performer> $Compensate_NoTerminateProcess_Service </Performer>
          </Execute>
        </Compensate>
</BusinessRules>
```

Fig. 6. The rule R1 of the RbBPDL purchase order after injecting the compensate code

```
<Compensate>
   <Discover>
        <Operation> $Subtitute_$Costume_Verification </Operation>
        < BusinessRegistryLocation > $UDDI_URL </ BusinessRegistryLocation >
        < QoS> $ Subtitute_$Costume_Verification
          <Quality name="cost" unit ="euro/acces" >
            <MaxValue> 5 </ MaxValue >
          </Quality>
        </ QoS>
   </ Discover >
</Compensate>
```

Fig. 7. Substitution code for the rule R1

The third alternative remedy can be the roll-backing that consist of, in BP-FAMA framework, the allowing the return to the rules in question in process model in the case of detecting an exception in the execution phase. For this reason, The BP-FAMA business process is translated into a new execution language called BPAEL. Indeed, this new language is entirely based on the BPEL standard, at which we add a new structured activity called "Rule". The latter allows identifying the business rules in a process BPAEL. In that way, the roll-backing to the question rule is possible in the case of occurring an exception in the execution phase.

Finally, note that, the live-lock exception handling code added into ending rule circuit (in the preceding example R_6) will remove all the previously data states saved during the various loop iterations. This is why the live-lock exception handling code is added only on to the starting and ending circuit rule.

6 Summary

In this paper we proposed to offer a self-healing execution of a business process within the BP-FAMA framework presented in [1]. Indeed, the BP-FAMA architecture

proposed in Figure 1 illustrates how the RbBPDL language is the core of this framework. RbBPDL uses ECA2PC paradigm to describe a business process using a set of business rules that are translated into a cause/effect graph. The analysis of this graph guarantees offering the self-healing systems by automatically discovering, diagnosing, and reacting to exceptions in order to drive the process execution towards a stable situation.

In the future, we aim to take into accounts the dynamic of the various process elements, by focussing more on the flexibility of business processes modeling. Indeed, we believe that a change in a process element may require changing other elements that are related to this element for the sake of maintaining the consistency of this process. For this reason, we aim in, our future work, to study the relationship between the rules in order to automate the flexibility modeling management.

References

1. Boukhebouze, M., Amghar, Y., Benharkat, A.: BP-FAMA: business process framework for agility of modelling and analysis. In: ICEIS: 10th International Conference on Enterprise Information Systems ICEIS 2008, Barcelona, SPA (2008)
2. OASIS: Business Process Execution Language for Web Services (BPEL4WS): Version 2.0. BPEL4WS specification report (2007)
3. The Workflow Management Coalition: Workflow Management Coalition Workflow Standard Process Definition Interface: XML Process Definition Language. The specification rapport (2005)
4. Pesic, M., van der Aalst, W.M.P.: A declarative approach for flexible business processes management. In: Eder, Dustdar (eds.), pp. 169–180 (2006)
5. Goedertier, S., Vanthienen, J.: Designing compliant business processes with obligations and permissions. In: Eder, Dustdar (eds.), pp. 5–14 (2006)
6. Object Management Group, Semantics of Business Vocabulary and Business Rules (SBVR) (2006), http://www.omg.org/spec/SBVR/1.0/PDF
7. Goedertier, R.H., Vanthienen, J.: EM-BrA2CE v0.1: A vocabulary and executionmodel for declarative business process modeling. FETEW Research Report KBI 0728, K.U.Leuven (2007)
8. Ganek, A.G., Corbi, T.A.: The dawning of the autonomic computing era. Technical report, IBM
9. Russell, N., van der Aalst, W.M.P., ter Hofstede, A.H.M.: Exception Handling Patterns in Process-Aware Information Systems. BPM Center Report BPM-06-04, BPMcenter.org (2006)
10. Subramanian, S., Thiran, P., Narendra, N.C., Mostefaoui, G.K., Maamar, Z.: Enhanced BPEL for Self-Healing Composite Web Services. In: IEEE International Symposium on Applications and the Internet (SAiNT 2008), Turku, Finland (2008) (published in IEEE LNCS)
11. Ouyang, C., Verbeek, E., van der Aalst, W.M.P., Breutel, S., Dumas, M., ter Hofstede, A.H.M.: WofBPEL: A tool for automated analysis of BPEL processes. In: Benatallah, B., Casati, F., Traverso, P. (eds.) ICSOC 2005. LNCS, vol. 3826, pp. 484–489. Springer, Heidelberg (2005)
12. Yang, Y., Tan, Q., Yu, J., Liu, F.: Transformation BPEL to CP-nets for verifying web services composition. In: The International Conference on Next Generation Web Services Practices, Korea (2005)
13. Wagner, G., Giurca, A., Lukichev, S.: Modeling Web Services with URML. In: Proceedings of Semantics for Business Process Management Workshop, Budva, Montenegro (2006)

Towards Flexible Inter-enterprise Collaboration: A Supply Chain Perspective

Boris Shishkov[1], Marten van Sinderen[2], and Alexander Verbraeck[3]

[1] TU Delft, Department of Systems Engineering/IICREST
Delft, The Netherlands
b.b.shishkov@tudelft.nl, b.b.shishkov@iicrest.eu
[2] University of Twente, Department of Computer Science
Enschede, The Netherlands
m.j.vansinderen@ewi.utwente.nl
[3] TU Delft, Department of Systems Engineering
Delft, The Netherlands
a.verbraeck@tudelft.nl

Abstract. Since neither uniformity nor pluriformity provide the answer to easing inter-enterprise collaborations, we address (inspired by relevant strengths of service-oriented architectures) the problem of supporting such collaborations from an infrastructure perspective. We propose architectural guidelines for interactively establishing a suitable inter-enterprise collaboration scheme, before the exchange of actual content takes place. The proposed guidelines stem from an analysis of some currently popular approaches concerning the achievement of inter-enterprise collaborations with ICT means. Taking into account the strong relevance of these issues to the Supply chain domain, we put our work in the Supply chain perspective. We also illustrate our architectural guidelines with an example from this domain. It is expected that the research contribution, reported in this paper, will be useful as an additional result concerning the (ICT-driven) inter-enterprise collaboration.

Keywords: Inter-enterprise collaboration, Service-oriented architectures, (non-) Standardized collaboration, Broker-mediated collaboration, Supply chain, Knowledge-based traceability.

1 Introduction

An *inter-enterprise collaboration* requires collaboration mechanisms that concern both *organizational aspects*, e.g. to agree on a joint process, and *technological aspects*, e.g. to enable the information exchange [7]. Depending on its role in such collaborations, an enterprise may need to exchange information with up to hundreds of other enterprises, as is often in a *supply chain* [3,28,8], for example. A Supply Chain (*SC*) collaboration can be considered as consisting of a number of bilateral collaborations where the enterprises involved in more than one bilateral collaboration are responsible for the (self-assumed or agreed-upon) coordination of these collaborations. Each bilateral collaboration is usually driven by a *collaboration contract* that describes the

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 513–527, 2009.
© Springer-Verlag Berlin Heidelberg 2009

'normal' scenario for service delivery between two enterprises as well as a number of foreseen 'exception' (deviation) scenarios.

An enterprise may encounter difficulties however, if it would need to replace a peer enterprise by a new one in an existing collaboration and/or start up a new collaboration. Such changes are not easy to achieve since new contracts and underlying mechanisms have to be established; moreover, accidentally introduced errors may proliferate via the enterprise to other collaborations. Inter-enterprise collaboration is facing therefore *inflexibility*, unacceptable *management overhead*, *error propagation*, high *latency*, and *inefficient use of resources* [4].

Standardizing the mechanisms for bilateral collaborations is not very helpful since the resulting standards must be all-encompassing (i.e., address all aspects of the collaboration that fulfil the requirements of as many enterprises as possible). Such standards would be therefore complex and voluminous, and also hard to agree upon and/or change [4,5,23]. Introducing broker systems that handle multiple specialized collaborations and provide bridges between them is not very helpful either. The reason for this is that full translation between collaboration mechanisms is expensive or sometimes even impossible, in which case human intervention is needed [21]. Moreover, broker approaches are inflexible since a change in one collaboration mechanism impacts all translations between this mechanism and the other mechanisms that the broker supports. Finally, it is often that enterprises do not trust a broker.

Considering *SOA* - Service Oriented Architecture [22] and adopting a SOA approach, in which each enterprise presents its collaboration capabilities as self-contained and loosely-coupled services, may be considered attractive for the following reasons:

- services are technology-agnostic, i.e. they may be implemented by an enterprise in any way without visibility for other enterprises that act as service users;
- services are self-describing and discoverable, i.e. they have descriptions stored in a registry that can be queried by service users;
- services are composable, i.e. orchestrations of services can be specified and executed resulting in 'higher-level' services for service users.

This certainly requires a distributed computing infrastructure, which is supported nowadays to some extent by Web services standards [17].

SOA has therefore some clear potential benefits, with an open question nevertheless: whether relatively simple services can be defined and described, such that suitable orchestrations of such services can realize long-running inter-enterprise collaborations with many enterprises involved. Another question is who would actually do the orchestration and provide the overall functionality.

For this reason, we expect that a SOA approach can be beneficial, however only if it is the case that the specific problems of inter-enterprise collaboration are adequately addressed, especially with regard to the SC domain.

Inspired by some recent related achievements [27], we propose in this paper *architectural guidelines* that concern inter-enterprise collaboration. In particular, it is envisioned interactively establishing a *suitable inter-enterprise collaboration scheme* before the exchange of actual content takes place. The scheme may specify: (i) collaboration protocols; (ii) content structures; (iii) orchestration processes. The successful negotiation of a collaboration scheme determines which partners can be

selected for specific business collaboration. Our proposed guidelines stem from an analysis of some currently popular approaches concerning the achievement of inter-enterprise collaborations with ICT means. Taking into account the strong relevance of these issues to the SC domain, we put our work in the SC perspective. We also illustrate our architectural guidelines with an example from this domain. It is expected that the research contribution, reported in this paper, will be useful as an additional result concerning the (ICT-driven) inter-enterprise collaboration.

This paper is further organized as follows. Section 2 contains an analysis of some currently popular inter-enterprise collaboration mechanisms. Section 3 considers (on this basis) related implications from the SC perspective. Section 4 presents our proposed architectural guidelines. Section 5 discusses an example of how they could be realized. Section 6 outlines some related work. Finally, Section 7 presents the conclusions.

2 Collaboration Approaches

Inter-enterprise collaborations are essential nowadays for enterprises in their aiming at delivering competitive services [18,29]. We can distinguish at least two collaboration perspectives, namely the 'informa perspective' (concerning information exchange and respectively the ability to formulate and interpret messages) and the 'performa perspective' (concerning the essential human ability of doing business, by engaging into commitments, either as performer or as addressee of a coordination act) [26]. In the following we focus mainly on the *informa perspective*: the exchange of messages with an agreed meaning that concern the achievement of some business effect. In addition, we explicitly consider *long-running collaboration propositions*, since usually inter-enterprise collaborations are evolving over a longer period of time. Such collaborations involve negotiations, commitments, contracts, shipping and logistics, tracking, varied payment instruments, deviation handling and customer satisfaction [21]. Furthermore, we assume that collaborations: (i) represent a function that is critical to the business and therefore should concern a shared business meaning; (ii) are usually evolving on top of a standards-based formal trading partner agreement, such as RosettaNet, Partner Interface Processes (PIPs) or ebXML Collaboration Protocol Agreements (CPAs); (iii) are driven by strict collaboration syntax and rules; (iv) define communications protocol bindings [6].

Inter-enterprise collaborations in general require distributed solutions of heterogeneous ICT systems.

Some of the currently popular inter-enterprise collaboration mechanisms are standardized (for example through the Electronic Data Interchange – EDI [21]) with limited application while others are rigid, hard-to-develop, non-standardized; neither of these mechanisms however fully responds to the increasing demands for flexible and adaptive collaboration, and according to some [19,18], the solution should be in the direction of service-oriented rule-based approaches. Others [7] claim that supporting such collaborations by means of brokers would respond better to these demands. Inspired by these and other studies, we propose the following classification of possible inter-enterprise collaboration mechanisms: (i) non-standardized bilateral collaboration; (ii) standardized bilateral collaboration; (iii) broker-mediated collaboration. Each of these collaboration types is elaborated below:

- *Non-standardized bilateral collaboration* consists of a set of pair-wise 'closed' collaborations, using rules which are private between each pair of enterprises and not approved by some standardization body, as illustrated in Figure 1*a*. Hence, if Enterprise A wants to collaborate with Enterprise B and Enterprise C, then collaboration schemes have to be agreed upon between each two parties separately. Problems thus are that each enterprise has to 'talk' many 'languages' and it is difficult to introduce a new enterprise in the collaboration because all others would have to 'learn' a new language, assuming that the new enterprise can not be forced to 'speak' languages already in use by the other enterprises.

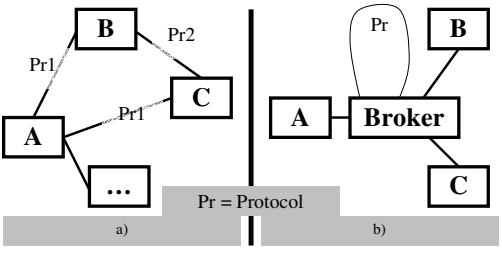


Fig. 1. Non-standardized collaboration (a), Broker-mediated collaboration (b)

- *Standardized bilateral collaboration* consists of a set of pair-wise collaborations driven by standards. Although standardization of inter-enterprise collaborations has often been proposed as a possible solution, standardization does not address all the observed difficulties, because of its decreasing flexibility. Next to that, it is hard for a single standard to deal with all types of deviations from the normal scenario, especially if the standard is to be used by thousands of enterprises with slightly different requirements. The early efforts in EDI have clearly shown this problem. Standards such as UN/EDIFACT [31] and ANSI ASC X12 [2] have become increasingly complex, while still not being able to address all potential problems in the collaboration process [19]. Due to its complexity, it is difficult to reach a global agreement on a standard, and on proposed changes and extensions. Standardization of inter-enterprise collaboration is a slow process [5], and changes are difficult to enforce and unpopular after the initial version, because there is already an installed base of users with dedicated software. In addition, such all-encompassing standards are hard to learn, not easy to adopt, and the software support is expensive.

- *Broker-mediated collaboration* makes use of brokers capable of 'understanding' multiple languages. There could be a central broker responsible for mediating all the 'conversions', as depicted in Figure 1*b*. Our assumption is that such a broker not only translates the syntax and semantic but also takes care of aligning the protocols, so that there are no process mismatches. Although protocol translations and syntax translations are possible, semantic problems often remain, and can lead to misinterpretation, source of failure in the collaboration, and a need for human interventions [21,32]. Being an intermediate party in the collaboration, the broker would often be insufficiently involved in dealing with deviations and errors, unless rigid rules are applied. The latter however would defy the flexibility and efficiency for which the broker was introduced. Another

concern is trust. Enterprises often do not trust a broker with important and commercial business content, thus reducing the applicability of brokers [19].

Hence, we argue that non-standardized bilateral collaborations are most secure, taking into account that it is not trivial for another entity to 'jump in'. The disadvantage here is the inflexibility with regard to new enterprises that might appear to be desired collaborators. This seems to be solvable by the enforcement of standards, which nevertheless points to many disadvantages at the same time, including the usual disagreement on who should introduce the standards and the insufficient security level. Brokers could be the desired mediators among enterprises, with a remaining question however whether everybody would trust a broker.

Applying SOA in each of the mentioned cases would result in collaborations running on top of a service infrastructure that takes care of basic interaction needs. For example: SOAP can be used for information exchange over HTTP and accessing a Web service; UDDI can be used for publishing and finding Web services. Ontologies that formalize relevant concepts, might also be used in each of the mentioned cases [1]. A broker, for instance, can be supported by a central ontology introduced. This would actually result in a single ('neutral') language, as illustrated in Figure 2 (as the figure shows, translations between languages A, B, and C come through the neutral language, x). Then, for each enterprise one would need only a translation between the enterprise language and the neutral language. Since it is impossible defining one ontology for the whole world, it would be necessary defining an 'upper ontology' (that is only reflecting the basic concepts and their relationships) and a specific ('domain') ontology, with a 'middle ontology' in between [16].

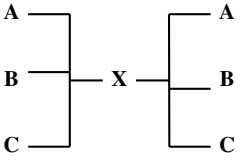


Fig. 2. Introducing a Central Ontology

And in the end, we would summarize the cross-cutting concerns that are to be taken into account when considering new architectural approaches for inter-enterprise collaboration: (i) flexibility in terms of collaborators and time; (ii) deviation handling; (iii) security and trust; (iv) cost; (v) change handling (how an enterprise would update its 'behavior' if there is a change in the environment).

3 Implications from the Supply Chain Perspective

In a SC, each organization typically needs to exchange information with several (possibly even thousands) other organizations (for example, if an organization has both a purchase and a sales function). Although SC standardization (ERP, CRM, workflow) has been proposed as a solution to the many-to-many problem, its effect is partial, just like in the general case and wrapping existing systems for talking to both enterprises seems only appropriate. The many-to-many problem has been hampering the successful

introduction of EDI for e-business, mainly because of increased implementation costs. Clearly, the readiness to be able to exchange information according to a number of protocols and workflow interaction patterns has a high price.

Information brokers have hence been introduced, 'acting between organizations', e.g. supporting the spot buy purchases of certain goods through trade exchanges [5]. Such brokers just combine information on supply and demand of goods without ever owning the goods themselves. The organizations still need to be able to exchange business messages with each other, in addition to 'finding' each other through the broker. Of course, brokers with more extensive functionalities exist as well: they take care of a larger part of the workflow. The network between organizations remains large however and a broker does not replace the majority of the connections. Organizations still want to be able to do business with each other without the broker (and avoid the additional payment for the broker's services).

A solution, widely considered, is replacing legacy applications by services, in the light of some claims that 'heavy-weight' monolithic systems could be usefully replaced by loosely-coupled (orchestrated) services providing adequate inter-organizational interactions [22,27]. These claims have not yet been properly proven in practice however. An important practical concern is the scalability issue: would it really be easy to scale up to realistic SCs [6]? It would only be easy, we argue, for organizations to exchange SC business messages if the syntactic, semantic, and pragmatic definition [16,26] of the required and offered services match, as it is depicted in Figure 3*a*.

With respect to this, we have several observations: (i) It is often hard for industry to agree on one representation standard and that's why a number of standards have been enforced over the last years by different software vendors trying to sell out their own solutions that are usually incompatible with other vendors' solutions [23]. (ii) Assuming that two organizations have different business logic, it would be challenging to channel all information through interfaces with the same definitions. Imagine, for example, that one of those organizations is introducing an additional service, enforcing thus the interface standards to accommodate the existence and functionality of that service for all organizations; (iii) With respect to service composition, a relevant question is who would actually 'orchestrate' the constituent services and provide the overall functionality.

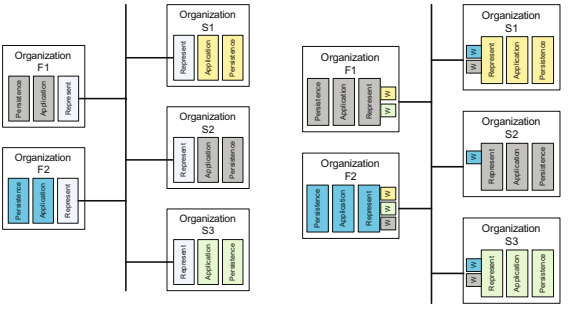


Fig. 3. Service-Oriented solutions - a) Idealistic services; b) Wrapped services

Indicative concerning the challenges mentioned above, is the incompatibility between industry solutions, such as Rosettanet [24], ebXML [9], and OpenXchange [20], not only at the representation level but also with respect to the business logic that constitutes the workflow. Hence, we would consider the popular claim that a possible solution to the pluriformity in the services domain is to introduce wrappers, keeping however the 'SOA context'. Figure 3*b* is presenting this solution in which wrappers would usefully enable services to 'talk' to many other implementations of the services in a service composition. This would mean, nevertheless, that a number of organizations should implement the same type of wrappers, which is costly and can easily lead to errors. The difficulty in making wrappers lies mainly in semantic and definitional differences, not so much in pure syntactic differences – these seem easily solvable, taking into account that current Web-service technology is standardized around XML and SOAP messages, with clear definition of the message formats in XML-schema or DTD files [34].

And in the end of this section, we will partially re-visit the cross-cutting concerns outlined at the end of the previous section; we would pay attention only to the SC-relevant ones: (i) Flexibility. In a spot-buy market, it is not known on beforehand with whom an enterprise will collaborate next. Therefore, the collaboration process has to be flexible, and should be able to work in many different configurations and under many different circumstances. (ii) Resilience to change. External requirements for the collaboration process are subject to continuous change. Policies and legal requirements change, and lead to needed adaptations of the process and of the information exchanged. These changes should not lead to drafting of a new version of the standard. (iii) Superfluous parts. A one-size-fits-all (all-encompassing) approach confronts the processes and users with a majority of fields and process steps that are not needed for the particular process, but rather for exotic versions of the process. This leads to overhead, errors, and implementation difficulties. (iv) Self-explanatory. When buying an IKEA cupboard [15], we don't expect everyone to know exactly how to put it together. Instructions are included and can be read on beforehand. In our information systems, even service-oriented ones, the instructions are not included and each system in the chain should know exactly how to handle each eventuality that can happen. Why not use the IKEA analogy for supporting inter-enterprise collaborations? This is a basis for the derivation of the main requirements (presented in the paragraph below) concerning the solutions to be proposed in Section 4.

REQUIREMENTS: (i) service orientation (as we have already concluded, a service-oriented solution would provide flexibility, re-use, and openness); (ii) protocol alignment (it is crucial that not only the syntax and semantics of the exchanged messages match, but also the message exchange protocols - this is pointing to needs for change impact analysis and related reasoning); (iii) no need for definitions of new standards (enforcement of yet another standard will not be easy); (iv) no broker-driven solutions (brokers are often not trusted as third parties).

4 Solution Directions

Concluding that neither uniformity nor pluriformity provide the answer to easing inter-enterprise collaborations, we have to look for another solution. When humans

exchange information in a complex setting, they discuss firstly the process, terms and conditions, after which the process is executed according to what has been agreed upon. This mechanism has been 'reflected' in workflow and orchestration languages describing a sequence of process steps that have to be executed between parties in order to implement an inter-enterprise collaboration; such languages have been analyzed by Honig [14]. Often, however, these workflows are defined once, and are not adapted to the properties of the particular process. There are also languages that describe a contract between parties, which specifies the conditions under which the exchange takes place, but they are usually quite static in nature [13]. A flexible variant that combines the dynamics of workflow and orchestration with the rigidness of the contracts (pre-conditions, post-conditions, time-outs, exception handling, and so on) would be attractive. First attempts to define languages describing the content of contracts that contain both content and workflow have been proposed already, e.g. the LinC language [14]. Still, these languages are not yet self-explanatory and do not contain their own 'manual' how to implement them.

In a sense, the exchange is a matter of matching the processes and information sent and received seen from one party with the processes and information received and sent by the other party.

In Figure 4, we see that Enterprise E1 expects to engage in a workflow of messages with another enterprise of the following sequence: (send A, receive B, send C, receive D, send E). Therefore, it looks for a partner that can interact in the following way: (receive A, send B, receive C, send D, receive E). When the two enterprises can agree on this workflow before starting to send and receive actual messages with content, i.e. agreeing on the protocol of interaction, they can be sure that the workflow can be continued from start to finish, and that the two enterprises will be able to make the deal they intended. To make this work, it is necessary that all issues mentioned above are adequately negotiated.

One of our proposals therefore would be considering an inter-enterprise modeling language that contains: (i) description of the normal workflow between parties; (ii) description of exceptional workflows between parties where needed; (iii) conditions on the workflow (e.g. timing, consistency, stop criteria); (iv) description of the content for each interaction step (using an existing ontology); (v) conditions on the content before and after each interaction step; (vi) instructions on how to handle the content in each instruction step (either computer readable or computer executable in case of a normal process, or human readable in case of error handling or escalation). Of course, such an inter-enterprise collaboration modeling language (to be addressed in more detail in further research) should again be described and formalized using another language. Furthermore, because we reason about content, an ontology is needed that describes the (part of the) world we are interested in. As we do not need one ontology, but we can use multiple ontologies and choose the one that is most appropriate for the problem at hand, it is not necessary to standardize this, which again helps with the flexibility demand. In realizing this we propose using the steps outlined in the following paragraphs.

First Step. Parties need to find each other. This is not much different from the discovery function in service-oriented architectures [17]. In many cases, parties will already know each other, and discovery is not a necessary function.

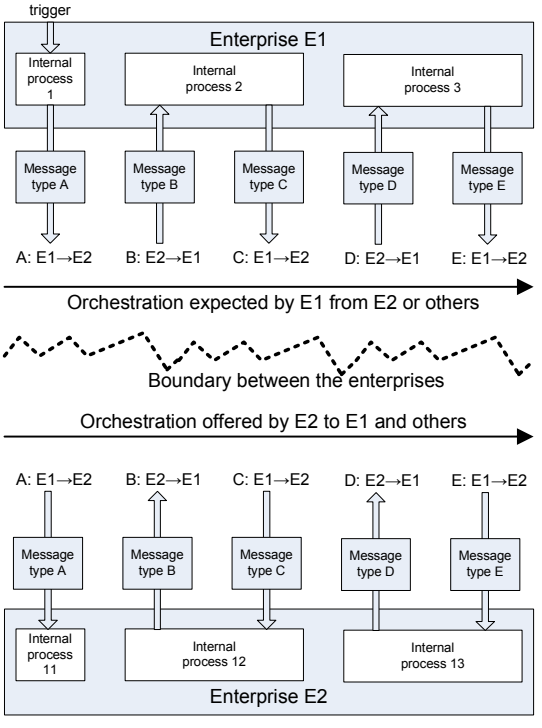


Fig. 4. A simplified example for an exchange between enterprise A and enterprise B that matches

Second Step. Matching the workflow (including deviations) and content (including ontology) between the enterprises. When enterprise A has defined a number of potential partners to work with, B_1, B_2, ..., B_n (in a simple bilateral agreement), a negotiation process is initiated that should identify whether each pair of enterprises can agree on joint content based on a joint ontology, and on a joint process. In a simple implementation, this matching process could be a simple one: enterprise A has a number of potential processes it could use (maybe a subset of an openly available set of potential inter-enterprise workflows), and each enterprise B_i has a set of processes as well. A match between the processes can establish whether they share a common workflow. If not, the negotiation with potential partner B_i can stop, and A can continue to negotiate with potential partner B_{i+1}. In a more complex solution, an algorithm can judge whether the differences are part of the critical part of the process or not, and can decide to use a partly overlapping process, or leave the decision to a human. In the end, this leads to the identification of zero or more partners to work with. If zero, the discovery phase can be redone or widened, or the failure can be escalated to a human who can decide on what to do. If more than one partner is found,

the best match can be chosen, or, in case of a spot buy market, the process can be continued with multiple partners.

Third Step. The process can be executed, following the agreed workflow; in some cases, executable components might be available to help with the execution of the process itself. These could be small services that are part of the workflow, and that can handle certain tasks. The fact that the executable components are part of the package, also could give the other partner(s) in the inter-enterprise collaboration some insight into how certain steps are executed. This could increase trust, and lead to a better acceptance of the inter-enterprise collaboration process. Furthermore, these components would reduce costs for the enterprises that implement the platform, especially if they are based on accepted standards, e.g. XML, SOAP, etc. [22].

In many cases, more than two enterprises would be involved in the information exchange. The orchestration can be done in exactly the same way, with increased complexity however, and more points of potential failure. Thus, we suggest considering the workflow as a transaction that can be rolled back completely, in case it does not succeed.

At a later stage, brokers could be added to this picture again, taking care of tasks that parties want to outsource (this is another role than that of the brokers discussed earlier) as part of their workflow process (payment through credit cards, certification of creditworthiness of partners, dealing with customs, and so on).

In responding to the trust concern however, a possible solution, especially relevant to the brokerage approach, would be in the direction of brokerage software that might be downloaded by each enterprise and translate within their boundaries the messages making them easily exchangeable with other enterprises using the same software.

Knowledge-Base Support. Matching the protocols, content structures, and orchestration of multiple partners, to find matches or near-matches, is considered to be a challenge. If we know for example that (i) Enterprise E_1 has dealt with Enterprise E_2 on orchestration O_3 for content types C_1, C_2 and C_3; (ii) Enterprise E_2 has dealt (at some point in time) with Enterprise E_3 using the same orchestration and content types; (iii) E_3 does not object to share this information with the partners of its partners (analogue to LinkedIn, FOAF, Amazon), then, if E_1 asks E_2 about suitable partners, E_2 would inform E1 about the existence of E_3. Such matching nevertheless requires complex *reasoning* related to tracing all relevant information.

We thus need to perform *knowledge-based traceability*, tracing back the previous collaborations of each of the enterprises; inspired from previous experience [27], we propose a traceability framework, whose technology-independent view is depicted in Figure 5.

As the figure suggests, the infrastructural support to inter-enterprise collaborations should include keeping track of previous collaborations. The relevant information should be stored accordingly in a knowledge-base that could be queried whenever an enterprise is about to launch a new collaboration, in order to have a basis for reasoning in support of the discovery of suitable partners. The reasoning results should not only be presented to the enterprise but they should also deliver adaptation instructions that concern the list of most appropriate enterprises that is to be made as a final result for the support of the templates choice.

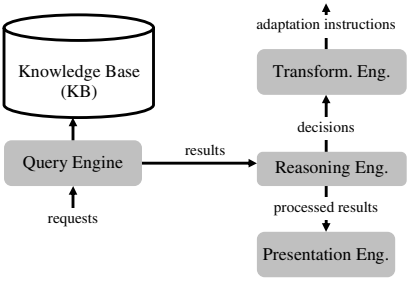


Fig. 5. Knowledge-based traceability – a technology-independent view

We will only partially illustrate some of these proposed solution directions in the following section, not going for a more thorough illustration for the sake of brevity.

5 Example

When two enterprises engage in an inter-enterprise collaboration, the overall pattern of collaboration involves a number of exchanges, e.g. when one enterprise buys a product at another enterprise, the minimal exchange taking place is depicted in Fig. 6:

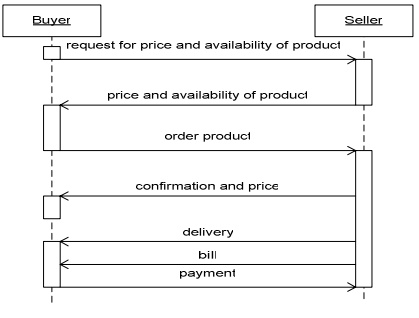


Fig. 6. Simple sequence of activities for buying a product

Of course, this process gets more complex when we have more than two parties involved and when we have a longer sequence of processes, or sub-processes. Furthermore, there are many points where this process can stop or time-out. The process might stop if the product is not available, or too expensive. A time-out occurs for instance when the intended seller does not answer the buyer in time. In that case, the buyer might look at another seller to see if this enterprise is more prompt, and whether the item is available there. If after that point in time the first seller would react, the process should prescribe what to do; discard the offer or take it into account anyhow. The seller also has a time-out on the offer; after some time the availability and price information does not hold anymore.

Let's call the buyer E_B and the seller E_S. The messages they try to exchange are the request M_R, availability and price M_A, order M_O, confirmation M_C, delivery M_D, bill

M_B, and payment M_P. There are also a number of protocols available to exchange messages, $P = \{p_i\}$, which is defined centrally and can be extended at any point in time. E_B can work with a subset of these protocols P_B, and E_S with a subset P_S. Furthermore, each of the message types $M = \{m_i\}$ has a definition, e.g. in an XML schema or DTD. In a simplified form, we could say that each message type consists of a sequence of name-class tupels $t_j = (n_j, c_j)$ where the name and class (type) are defined in an ontology. Thus, $m_i = \{t_{i,1}, t_{i,2}, ..., t_{i,n}\}$. An orchestration O is defined by a sequence $\{(M_1, d_1), (M_2, d_2) ..., (M_n, d_n)\}$, where $d_i \in \{in, out\}$ indicates the direction.

To find out whether we have a match, there are a number of steps. First, E_B sends P_B to E_S. E_S calculates $P_B \cap P_S$ and sends back the result $P_{B\cap S}$. If $P_{B\cap S}=\varnothing$, a match cannot be made and E_B will have to look for another supplier. If a protocol can be chosen, the next step is to see if E_B and E_S can work with the same message types. E_B sends each of the message types that it has in its sequence $m_{B,i}$ to E_S. E_S will match this with its internal list of available message types, and makes comparisons for each name-class tupel $t_j = (n_j, c_j)$ to see if it matches. A match in this case could also be formed by a subset of the range for the values. For each $m_{B,i}$ for which a matching $m_{S,k}$ has been found, E_S will send back the $m_{S,k}$ for inspection by E_B. If E_B agrees on the match as well, the orchestration negotiation can start.

For the orchestration, E_B will send its request $O_B = \{(M_A, out), (M_O, in), (M_C, out), (M_D, in), (M_B, in), (M_P, out)\}$ to E_S. E_S will look in its repository of available orchestrations around these messages to see if it has a matching protocol, replacing d_i in each (M_i, d_i) by $\neg d_i$. When it finds one, it sends back the confirmation to E_B, which can choose to work with E_S now to work on an actual collaboration where the messages are exchanged in the indicated order governed by O_B, with message content defined by $M_{B\cap S}$, and using a network protocol $p_i \in P_{B\cap S}$.

6 Related Work

The key issue of inter-enterprise collaboration is *interoperability*. The trend to globalized markets has painfully made clear that many enterprise systems are not designed to interoperate with other systems of other enterprises [33]. Most of the problems emerge from proprietary development or extensions, unavailability or oversupply of standards, and heterogeneous hardware and software platforms. A major challenge is to achieve and sustain interoperability in the face of planned and spontaneous changes, with proper alignment between and integrity of the business and technology levels [29]. Facing this challenge and developing solutions that not only solve current enterprise interoperability problems but also create new business opportunities, is one of the focal points of the European Commission [11,12].

Our contribution in this paper focuses on the exploration of new architectural patterns for collaboration (or interoperability). We explained the principle steps in the approach that embodies these patterns, but we are well aware that many issues need to be addressed and technological support need to be developed in order to make this work in practice.

To take some inspiration in this, we turn to some recent results which directly or indirectly relate to the problems we are addressing in this research. The mentioned results are in general in three directions: (i) Cross-organizational collaboration; (ii)

Service design and execution; (iii) Interoperability architectures. We will only mention several relevant recent examples of related work, one in each of these three directions, for brevity.

With regard to cross-organizational collaboration, Schroth has proposed a service-oriented reference architecture for business media that overcomes the typical B2B software drawbacks, by considering four main views, namely community (structural organization), process (process-oriented organization), services and infrastructure [25]. Concerning services, a service bus actually enables and facilitates interactions on the basis of operational as well as coordination services.

Concerning service design and execution, some new service models have been introduced, such as the one proposed by Esper [10], which is essentially driven by business interoperability constraints.

As it concerns interoperability architectures, a meta-model has been proposed by Ullberg [30], which meta-model could support the creation of enterprise architecture models amenable to service interoperability analysis. This can be represented using influence diagram with attributes affecting service interoperability.

All this experience has inspired us mostly indirectly for our architectural contribution, and the analysis presented in this section further justifies in our opinion the claim that current approaches (including standard and broker-based) are insufficiently capable of facilitating inter-enterprise collaborations.

7 Conclusions

In this paper, we have presented solution directions that concern inter-enterprise collaboration and in particular, the desired capability of enterprises changing flexibly their (supply-customer) networks. We propose architectural guidelines for establishing the inter-enterprise collaboration protocols, content structure, and orchestration (the inter-enterprise process) *before* the exchange of actual content takes place.

Distinctive features of the proposed guidelines are the close-to-real-life service-oriented collaboration and related to this: no need for support through all-encompassing standards and/or undesired third parties (brokers). Moreover, with regard to multiple partner's needs to find matches or near matches, we have addressed the challenge of matching the protocols, content structures, and orchestration of multiple partners.

To further this research, we plan to: (i) elaborate more on the solution we have proposed, by considering functional and informational architectural issues; (ii) propose an implementation related to the suggested knowledge-based traceability, possibly using Prolog.

Acknowledgements. This work has been supported by the Systems Engineering Dept. at TU Delft and by the Freeband A-MUSE project (http://a-muse.freeband.nl). Freeband is sponsored by the Dutch government under contract BSIK 03025.

References

1. Alonso, G., Casati, F., Kuno, H., Machiraju, V.: Web Services, Concepts, Architectures and Applications. Springer, Heidelberg (2004)
2. ASC X12 2008, `http://www.x12.org`

3. Bowersox, D.J., Closs, D., Cooper, M.B.: Supply Chain Logistics Management. McGraw-Hill, USA (2002)
4. Boyson, S., Corsi, T.M., Dresner, M.E., Harrington, L.H.: Logistics and the Extended Enterprise. John Wiley, USA (1999)
5. Boyson, S., Corsi, T.M., Verbraeck, A.: The E-supply Chain Portal: A Core Business Model. Logistics and Transportation Review Part E 39, 175–192 (2003)
6. Boyson, S., Harrington, L.H., Corsi, T.M.: In Realtime: Managing the New Supply Chain. Greenwood Publishers / Praeger (2004)
7. Camarinha-Matos, L.M. (ed.): Collaborative Business Ecosystems and Virtual Enterprises. Kluwer Academic Publishers, Dordrecht (2002)
8. Corsi, T.M., Boyson, S., Verbraeck, A., Van Houten, S.P.A., Han, C., MacDonald, J.: The real-time global supply chain game: New educational tool for developing supply chain management professionals. Transportation Journal 45(3), 61–73 (2006)
9. ebXML 2008, http://www.x12.org
10. Esper, A., Sliman, L., Badr, Y., Biennier, F.: Towards Secured and Interoperable Business Services. In: Mertins, K., Ruggaber, R., Popplewell, K., Xu, X. (eds.) Enterprise Interoperability III. Springer, Heidelberg (2008)
11. European Commission: Enterprise interoperability research roadmap. Version 4.0 (July 2006), ftp://ftp.cordis.europa.eu
12. European Commission: Unleashing the potential of the European knowledge economy. Value proposition for enterprise interoperability. Version 4.0 (January 2008), ftp://ftp.cordis.europa.eu/pub/ist/docs/ict-ent-net/isg-report-4-0-erratum_en.pdf
13. Gelernter, D., Carriero, N.: Coordination Languages and Their Significance. Communications of the ACM 35(2), 97–107 (1992)
14. Honig, J.: Towards On-line Logistics: the LinC Interaction Modeling Language, PhD Thesis. TU Delft Press (2004)
15. IKEA 2008, http://www.ikea.com
16. Liu, K.: Semiotics in Information Systems Engineering. Cambridge University Press, Cambridge (2000)
17. Newcomer, E.: Understanding Web Services, XML, WSDL, SOAP and UDDI. Addison-Wesley, Boston (2002)
18. Orriens, B.: Modeling The Business Collaboration Context, Technical Report, Tilburg University, The Netherlands (2006)
19. Orriens, B., Yang, J.: Establishing and Maintaining Compatibility in Service-Oriented Business Collaboration. In: 7th International Conference on Electronic Commerce (2005)
20. OXpedia 2008, http://www.open-xchange.com/en/oxpedia
21. Papazoglou, M.: The World of e-Business: Web-Services, Workflows, and Business Transactions. In: Bussler, C.J., McIlraith, S.A., Orlowska, M.E., Pernici, B., Yang, J. (eds.) CAiSE 2002 and WES 2002. LNCS, vol. 2512, pp. 153–173. Springer, Heidelberg (2002)
22. Papazoglou, M.P.: Web Services: Principles and Technology. Addison-Wesley, Reading (2007)
23. Poirier, C.C.: Advanced supply chain management. Berrett-Koehler Publishers, San Francisco (1999)
24. Rosettanet 2008, http://www.rosettanet.org/cms/sites/RosettaNet
25. Schroth, C.: A Service-oriented Reference Architecture for Organizing Cross-Company Collaboration. In: Mertins, K., Ruggaber, R., Popplewell, K., Xu, X. (eds.) Enterprise Interoperability III. Springer, Heidelberg (2008)

26. Shishkov, B., Dietz, J.L.G., Liu, K.: Bridging the Language-Action Perspective and Organizational Semiotics in SDBC. In: ICEIS 2006 8th Int. Conf. on Enterprise Inf. Systems (2006)
27. Shishkov, B., Van Sinderen, M.J., Quartel, D.: SOA-Driven Business-Software Alignment. In: ICEBE 2006, IEEE International Conference on e-Business Engineering (2006)
28. Simchi-Levi, D., Kaminsky, P., Simchi-Levi, E.: Designing & Managing the Supply Chain, 2nd edn. McGraw-Hill, USA (2003)
29. van Sinderen, M.J., Johnson, P.C., Kutvonen, L.: Report on the IFIP WG5.8 Int. Workshop on Enterprise Interoperability (IWEI 2008). In: ACM SIGMODD Record (2008) (to appear)
30. Ullberg, J., Lagerstrom, R., Johnson, P.C.: Enterprise Architecture: A Service Interoperability Analysis Framework. In: Mertins, K., Ruggaber, R., Popplewell, K., Xu, X. (eds.) Enterprise Interoperability III. Springer, Heidelberg (2008)
31. United Nations: Electronic Data Interchange for Administration, Commerce and Transport (2003), http://www.un.org/Pubs/whatsnew/e99v06.htm
32. Van de Kar, E.A.M., Verbraeck, A.: Designing Mobile Service Systems. Research in Design Series, vol. 2. IOS Press, Amsterdam (2007)
33. Wang, H., Zhang, H.: Enabling Enterprise Resources Reusability and Interoperability Through Web Services. In: ICEBE 2006, IEEE Int. Conf. on e-Business Engineering (2006)
34. XML 2008, http://www.w3.org/XML

A Model-Based Tool for Conceptual Modeling and Domain Ontology Engineering in OntoUML

Alessander Botti Benevides and Giancarlo Guizzardi

Ontology and Conceptual Modeling Research Group (NEMO),
Computer Science Department, Federal University of Espírito Santo (UFES)
Av. Fernando Ferrari, nº 514, Goiabeiras, Vitória (ES), Brazil
{abbenevides,gguizzardi}@inf.ufes.br

Abstract. This paper presents a Model-Based graphical editor for supporting the creation of conceptual models and domain ontologies in a philosophically and cognitively well-founded modeling language named OntoUML. The Editor is designed in a way that, on one hand, it shields the user from the complexity of the ontological principles underlying this language. On the other hand, it reinforces these principles in the produced models by providing a mechanism for automatic formal constraint verification.

Keywords: Ontology Engineering, Conceptual Modeling.

1 Introduction

Throughout the last years, ontologies have increasingly been applied in Computer Science. They have been a topic of research in Artificial Intelligence (AI) since the late seventies and more recently, in Software Engineering (SE). On the one hand, in the former, ontologies have been used as a knowledge representation technique to convey domain terminologies (*e.g.*, Description Logic T-Boxes) to be particularized as facts (*e.g.*, Description Logic A-Boxes) for serving to reasoning purposes. In the latter, on the other hand, ontologies have been mainly recognized as comprising a technique for developing enhanced domain-specific conceptual models.

There are two common trends in the use of ontologies in these two areas: (i) firstly, ontologies are always regarded as an explicit representation of a shared conceptualization, *i.e.*, a concrete artifact representing a model of consensus within a community and a universe of discourse. Moreover, in this sense of a reference model, an ontology is primarily aimed at supporting semantic interoperability in its various forms (*e.g.*, model integration, service interoperability, knowledge harmonization); (ii) secondly, the discussion regarding representation mechanisms for the construction of domain ontologies is, in both cases, centered on computational issues, not truly ontological ones. On one side, the AI community values representation languages which prime for computational tractability [1]. On the other side, the SE community is mostly concerned with committing to the use of standardized languages such as the Unified Modeling Language (UML) [2], and with producing ontology representations that facilitates the mapping to specific implementation environments (*e.g.*, Object-Oriented Frameworks [3]). Now, an important aspect to be highlighted is the

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 528–538, 2009.
© Springer-Verlag Berlin Heidelberg 2009

incongruence between (i) and (ii). As shown, for instance in Guizzardi [4], in order for an ontology to be able to adequately support (i), it should be constructed using an approach that explicitly takes into account a dimension which is neglected in (ii), namely, the use of foundational concepts that take truly ontological issues seriously.

In pace with Degen *et al.* [5], we argue that "every domain-specific ontology must use as framework some upper-level ontology". This claim for an upper-level (or foundational) ontology underlying a domain-specific ontology is based on the need for fundamental ontological structures, such as theory of parts, theory of wholes, types and instantiation, identity, dependence, unity, etc, in order to properly represent reality. From an ontology representation language perspective, this principle advocates that, in order for a modeling language to meet the requirements of expressiveness, clarity and truthfulness in representing the subject domain at hand, it must be an ontologically well-founded language in a strong ontological sense, *i.e.*, it must be a language whose modeling primitives are derived from a proper foundational ontology [6], [7].

An example of a general conceptual modeling and ontology representation language that has been designed following these principles is the version of UML proposed in [8]. This language (later termed OntoUML) has been constructed in a manner that its metamodel reflects the ontological distinctions prescribed by UFO (Unified Foundational Ontology). Moreover, formal constraints have been incorporated in this language's metamodel in order to incorporate the formal axiomatization in UFO. Therefore a UML model that is ontologically misconceived taking UFO into account is syntactically invalid when written in OntoUML.

Although this approach has been able to provide mechanisms for addressing a number of classical conceptual modeling problems [9], and the language has been successfully employed in application domains [10], [11], there was still no tool support for building and validating conceptual models and domain ontologies constructed using OntoUML. The main contribution of this paper is thus to present a Model-Based OntoUML Graphical Editor with support for automatic model checking in face of ontological constraints. The binaries and source code files for the editor are available at http://code.google.com/p/ontouml.

A snapshot of the main section of the editor is shown in Fig. 1 below. As one can see, there is a tool bar on the right side, where the user can drag and drop model elements on the left panel.

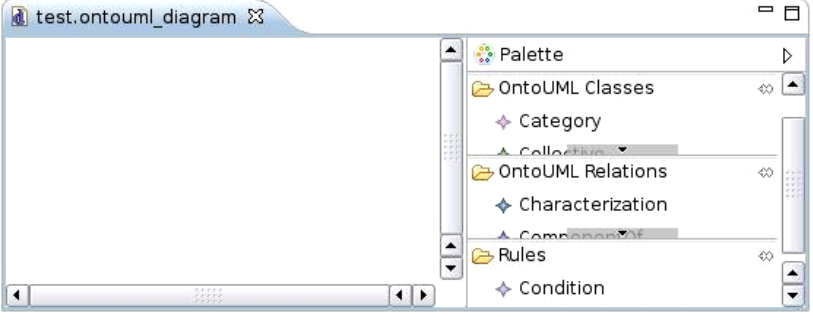


Fig. 1. Snapshot of the editor

The remainder of this paper is structured as follows. Section 2 presents models that illustrates main concepts of UFO, represented using OntoUML, and validated on the editor by automatically processing integrity and derivation rules that represent ontological constraints over the produced models. Section 3 briefly elaborates on the metamodeling and implementation technologies used in the construction of the editor. Section 4 presents some related work. Section 5 presents some final considerations.

2 Presentation of the Editor

In this section, we illustrate the support provided by the editor for automatically checking integrity constraints and deriving information in models. Integrity constraints are inspected via two different mechanisms named *Live Validation* and *Batch Validation*. In order to illustrate these features, let us make use of a simple domain model of car dealing. This simple universe of discourse is comprised of concepts such as Person, Car, CarCustomer, CarSupplier, Organization, Purchase and car parts (*e.g.*, Engine, Chassis). In the following we briefly exemplify how the editor can assist the user in the construction of a simple conceptual model in this domain.

2.1 Live Validation

In this conceptualization, Person would typically be modeled in OntoUML as a class with a <<kind>> stereotype, and a CarCustomer would be modeled as a class with a <<role>> stereotype, as is shown in Fig. 2. In OntoUML, the <<kind>> stereotype is used to represent the UFO Kind category, and the <<role>> stereotype represents the UFO Role category. In order to explain these UFO categories, we have to describe some more categories, which are represented on the excerpt of the UFO taxonomy in Fig. 3.

As we can see, Kinds and Roles are Entities, where Entity is the higher UFO category. Entity can be distinguished in Universal and Individual, where Individuals are entities that exist in reality possessing a unique identity, and Universals, conversely, are space-time independent pattern of features, which can be realized in a number of different individuals. In its turn, Universals can be distinguished in Monadic Universal and Relation (entities which glue together other entities). Within the category of Monadic Universal, in order to show the differences between Substance Universal and Relator Universal, we need to explicate what are Substances and Moments.

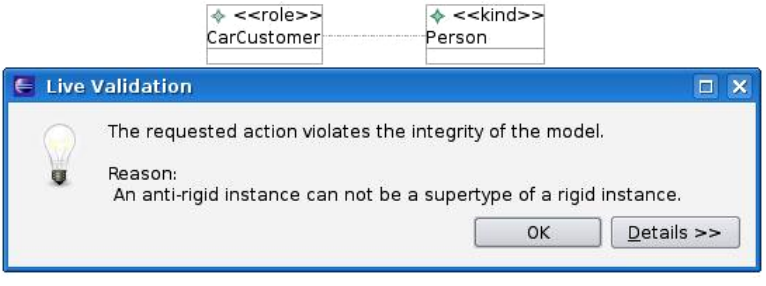


Fig. 2. Live validation example

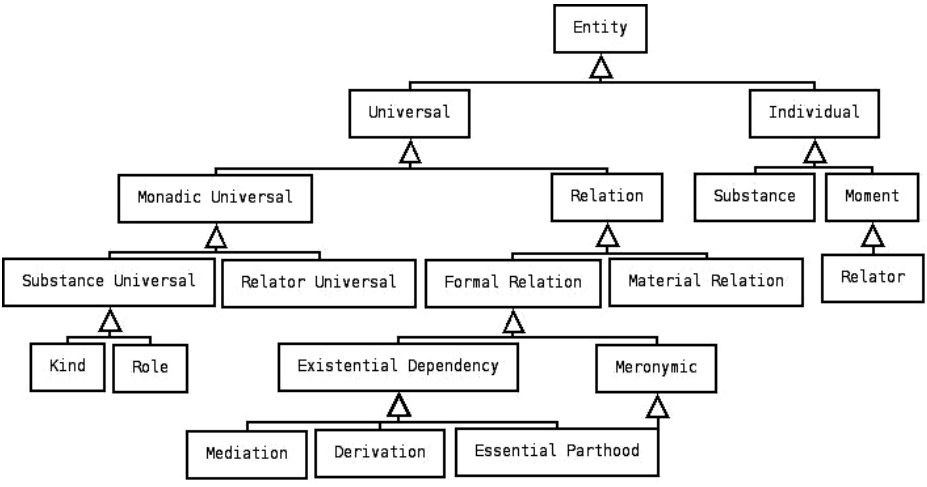


Fig. 3. Excerpt of UFO taxonomy [8]

Substances are existentially independent individuals, *i.e.*, there is no Entity x that must exist whenever a Substance y exists. Examples of Substances include ordinary mesoscopic objects such as an individual person, a house, a hammer, a car, but also the so-called *Fiat Objects* such as the North-Sea and its proper-parts, postal districts and a non-smoking area of a restaurant. Conversely, a Moment is an individual which can only exist in other individuals. Typical examples of moments are: a color, a connection, a purchase order.

So, a Substantial Universal is a universal whose instances are Substances (*e.g.*, the universal Person or the universal Apple). While, a Relator Universal is a universal whose instances are individual relational moments (*e.g.*, the particular enrollment connecting John and a certain University is an instance of the universal Enrollment).

We need to define some formal notions (rigidity and anti-rigidity) to be able to make further distinctions within Substance Universal. Definition 1 (Rigidity): A universal U is rigid if for every instance x of U, x is necessarily (in the modal sense) an instance of U. In other words, if x instantiates U in a given world w, then x must instantiate U in every possible world w'. ■ Definition 2 (Anti-rigidity): A universal U is anti-rigid if for every instance x of U, x is possibly (in the modal sense) not an instance of U. In other words, if x instantiates U in a given world w, then there must be a possible world w' in which x does not instantiate U. ■ [8].

A Substantial Universal which is rigid is named here a Kind. In contrast, an anti-rigid substantial universal is termed a Role. The prototypical example highlighting the modal distinction between these two categories is the difference between the universal (Kind) Person and the (Role) universal CarCustomer, both instantiated by the individual John in a given circumstance. Whilst John can cease to be a CarCustomer (and there were circumstances in which John was not one), he cannot cease to be a Person. In other words, in a conceptualization that models Person as a Kind and CarCustomer as a Role, while the instantiation of the role CarCustomer has no impact on the identity of an individual, if an individual ceases to instantiate the Kind Person,

then it ceases to exist as the same individual. Moreover, [9] formally proves that a rigid universal cannot have as its superclass an anti-rigid one. Consequently, a Role cannot subsume a Kind in our theory.

Now, as discussed in [9], a common mistake in conceptual modeling is the use of subtyping to represent alternative allowed types, *i.e.*, alternative types that supply players for a given role. In this particular case, suppose that the user attempts to represent that instances of Person are possible players of the role CarCustomer, by using subtyping. In other words, the user tries to model a Kind Person as a subtype of the Role CarCustomer. If allowed, this would not be an ontologically correct model, since it is not the case that every instance of Person is a CarCustomer, and since a Person cannot cease to be a Person but it can cease to be a CarCustomer. When attempting to create this ontologically incorrect model with the editor presented here, an integrity constraint is violated. As consequence, the editor ignores the corresponding model updating action and prompts a live validation pop-up that alerts the user of his attempt of creating an invalid model. The validation pop-up resulting from this example is shown in Fig. 2.

2.2 Deriving Model Information

In order to represent the relation between CarCustomer and Person, one should model CarCustomer as a Role played by Person in a certain context, where he buys a Car from a CarSupplier. Analogously, one should model CarSupplier as a Role played by an Organization when selling a Car to a CarCustomer. This context is materialized by the Material Relation *purchases* (represented as the <<material>> stereotype in OntoUML), which is in turn, derived from the existence of the Relator Universal Purchase (<<relator>>). In other words, we can say that a particular customer x purchases a particular car y from a particular supplier z iff there is a Purchase which mediates x, y and z. This situation is illustrated in Fig. 4. The mediation formal relations between the Relator Purchase and the Roles CarCustomer and CarSupplier are responsible for the existence of the derived Material Relation *purchases* that hold between CarCustomer and CarSupplier. Thus, the cardinality restrictions of the purchases relation can be

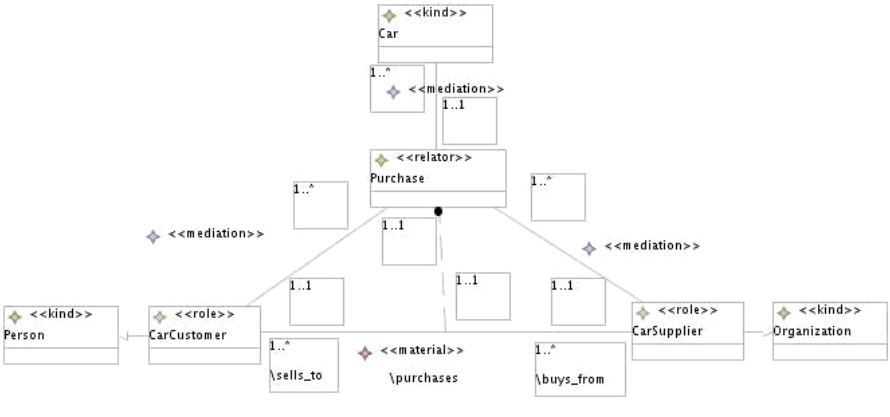


Fig. 4. Example of derivation of information

systematically calculated from these associations as exemplified in Fig. 5 below. The derivation of purchases from the mediation relations is represented by a Derivation association (pictured as a dashed line association between purchases and Purchase, where there is a black circle), which also have its cardinalities systematically calculated.

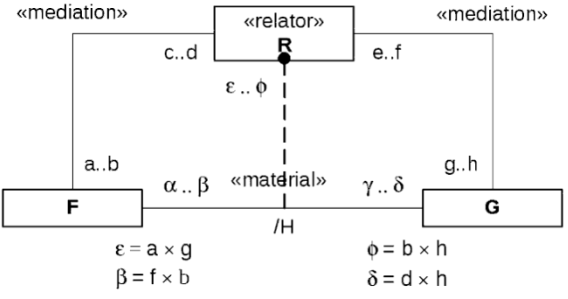


Fig. 5. Cardinality derivation

In order to better explain what is a Material Relation, a Mediation relation (<<mediation>>) and a Derivation relation, we need to describe more categories represented in Fig. 3. The Relation category is differentiated in *Formal Relation* and *Material Relation*, where Formal Relations are relations that hold between two or more entities directly, without any further intervening individual, and Material Relations, conversely, do need an intervening individual. Examples of formal and material relations are older_than and purchases, respectively. The notion Formal Relation is further differentiated here in Existencial Dependency and Meronymic, where the former represents existentially dependent associations and the latter represents part-whole relations. For now, we can consider two types of existentially dependent Formal Relations: Mediation and Derivation. A Mediation relation is a relation that holds between a Substantial Universal and Relator Universal. Mediation and Relator Universal are the basis for defining Material Relations. In order to a Material Relation M_1 hold between two Substantial Universals S_1 and S_2, there must exists at least two Mediation relations (M_2 and M_3) and one Relator R, such that M_2 holds between S_1 and R and M_3 holds between S_2 and R. The Derivation relation is a relation between a Material Relation M_1 and the Relator Universals on which M_1 depends [8].

2.3 Batch Validation

A more complete version of a model in this domain is shown in Fig. 6, which represents some of the parts that compose a Car. In this figure, it is represented that a Car is composed of one CarEngine. However, part-whole relations must obey the so-called Weak Supplementation axiom, which, in simple words, states that in order to be a whole, an entity must have at least two disjoint parts. Therefore, to satisfy this axiom, if a Car is composed of one and only one Engine, it must also have another car component as a part. Now, differently from the Person-CarCustomer subtyping example discussed above, the lack of a second part represented in the model that would

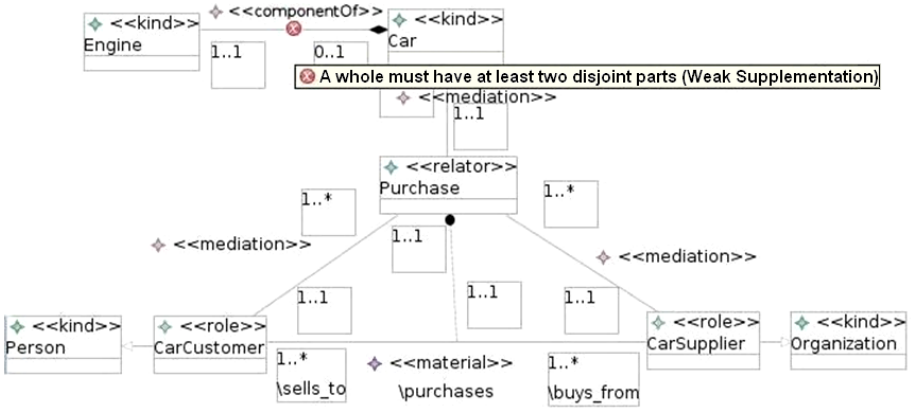


Fig. 6. Batch validation example

meet the requirement posed by the Weak Supplementation axiom can be due to a momentary incompleteness of the model. In other words, after the part-whole relation between Car and CarEngine is represented, the user can still include information in the model that will prevent this model from being considered ontologically inconsistent. As this example shows, there are validation actions that should only be performed by the tool once the user deems suitable. Now, as illustrated in Fig. 6, if this model is validated with the presented information, the editor prompts to the user that, in that form, the model is considered incorrect. Furthermore, the editor informs the user by highlighting the source and reason of inconsistency in the model.

A possible solution to this issue is to represent that a Car is composed of something more than an Engine, *e.g.*, a Chassis. Fig. 7 depicts this alternative representation where a car is composed of one and only one Engine and an essential unique Chassis, where the "essential" tag in this part-whole relations means that the whole is existentially dependent of the part [8].

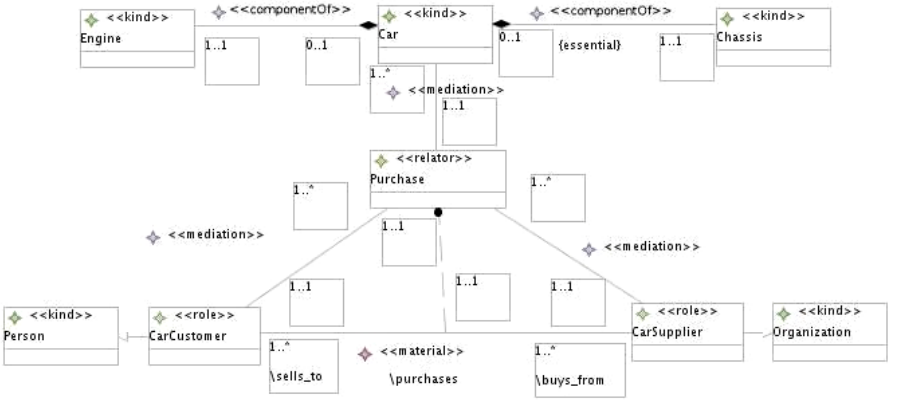


Fig. 7. A possible solution to correct the model pictured in Fig. 6 above

3 The Architecture and Implementation of the Editor

The architecture of the editor presented here has been conceived to follow a Model-Driven Approach. In particular, we have adopted the OMG MOF (Meta-Object Facility) metamodeling architecture [12]. In order to describe constraints in UML/MOF (meta)models, the OMG also proposes the declarative formal language OCL (Object Constraint Language) [13]. On the formalization of the OntoUML profile we have used OCL expressions mainly to: define how derived attributes/associations get their values; define default values of attributes/associations, *i.e.*, define their initial values; specify query operations and specify invariants, *i.e.*, integrity constraints that determine a condition that must be true in all consistent system states.

The complete implementation of the OntoUML profile as a MOF metamodel is reported in [14]. The same reference also describes the full set of OCL expressions including: 8 OCL expressions to specify derivation rules; 145 OCL expressions to define default values; 13 OCL expressions to specify operations created to support some OCL derivation rules and invariants, and 69 invariants to model the constraints stated on the OntoUML profile [8]. An example of an OCL invariant representing the essential parthood axioms described in OntoUML is shown in the code below. One can notice that in this expression the modal existential dependence constraint of essential parthood from UFO is emulated via the existence condition (lower cardinality ≥ 1) plus the immutability constraint (isReadOnly = true).

```
context Meronymic

inv: if (self.isEssential = true) then self.target->
forAll(x | if x.oclIsKindOf(Property) then
((x.oclAsType(Property).isReadOnly = true) and
((x.oclAsType(Property).lower >= 1)) else false endif)
else true endif
```

In terms of implementation technology, the editor has been implemented using a number of plug-ins that supports graphical editor development in the context of the Eclipse IDE (Integrated Development Environment) [15]. For the creation of the OntoUML metamodel, we have used the Eclipse Modeling Framework (EMF) [16], [17] plug-in. This plug-in provides its own metamodeling language named ECore, which asides from few (mostly terminological) differences is equivalent to the EMOF (Essential MOF) language (a subset of the complete MOF 2.0 language) [12]. The EMF together with the Model Development Tools (MDT) [18] plug-in allows for the creation and validation of ECore models with embedded OCL constraints. Finally, to build the graphical interface of the editor, we have used the Graphical Modeling Framework (GMF) [19] plug-in. The GMF provides a high-level description of visual representations to support transformation to a set of java classes for the graphical editor using a Model-View-Controller (MVC) architecture. This process is schematically summarized in Fig. 8 below.

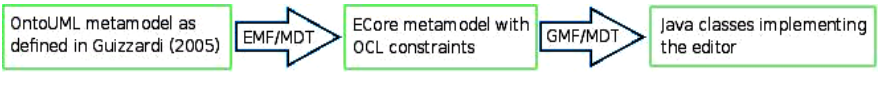


Fig. 8. Tool generation overview

4 Related Work

As far as we know, there is no other tool for OntoUML. However, there are other editors that support philosophically well-founded languages and methodologies such as OntoClean [20], as well as tools based on upper-level ontologies as SUMO (Suggested Upper Merged Ontology) [21], SUO (Standard Upper Ontology) [22] and the Differential Semantics theory. For instance, Protégé [23] is a free open-source tool which supports OntoClean and SUMO. AEON (Automatic Evaluation of ONtologies) [24] is an open-source tool, which allows applying OntoClean to evaluate ontologies. Visual Ontology Modeler [25] is an editor that includes a library of ontologies that represent SUO. DOE (Differential Ontology Editor) [26] is a freeware ontology editor which allows the user to build ontologies according to the Differential Semantics theory. Sigma [27] is a free open-source knowledge engineering environment for theories in first order logic (FOL), which is optimized for SUMO.

5 Final Considerations

The need for using ontologically well-founded languages for conceptual modeling, in general, and domain ontologies, in particular, has increasingly been recognized in the literature. This is often a result of interoperability concerns and the unsuitability of lightweight representation languages in addressing these issues. Despite that, these languages are still not broadly adopted in practice. One of the main reasons is the need for high-level expertise in handling the philosophical concepts underlying them. Indeed, the dissemination of formal method techniques requires convincing industries and standardization bodies that such techniques in fact can improve development. In this way, design support tools are one of the key resources to foster their adoption in practice [28].

In this paper, we present an Eclipse-based graphical editor which aims at fulfilling the gap of tool support for one particular theoretically well-founded representation language, namely, OntoUML. Underlying this editor there is an implementation of the OntoUML metamodel proposed by Guizzardi [8] by using MDA (Model-Driven Architecture) technologies, in particular, the OMG MOF (Meta-Object Facility) and OCL (Object Constraint Language). Moreover, by representing UFO categories and axiomatization in the language metamodel, the complexity of these foundational issues is hidden from the user while still constraining him to produce ontologically sound models.

As a final remark, the promotion of a language such as OntoUML for domain engineering does not eliminate the need for codification languages such as OWL, DLR_{us}, Alloy or F-Logic, to cite just a few examples. In pace with the meaning independence principle defended by Guizzardi and Halpin [29], we adopt the view that these classes of languages are (and should be) meant to be used for different purposes and in different phases on an ontology engineering process.

References

1. Levesque, H.J., Brachman, R.J.: Expressiveness and Tractability in Knowledge Representation and Reasoning. In: Computational Intelligence, vol. 3, pp. 78–93 (1985)
2. Object Management Group (OMG): UML 2.0 Superstructure Specification, Doc.# ptc/03-08-02 (2003)
3. Falbo, R.A., Guizzardi, G., Duarte, K.C.: An Ontological Approach to Domain Engineering. In: ACM XIV International Conference on Software Engineering and Knowledge Engineering (SEKE 2002), Ischia, Italy (2002)
4. Guizzardi, G.: The Role of Foundational Ontology for Conceptual Modeling and Domain Ontology Representation. In: 7th International Baltic Conference on Databases and Information Systems, Vilnius, Lithuania (2006)
5. Degen, W., Heller, B., Herre, H., Smith, B.: GOL: Toward an Axiomatized Upper-level Ontology. In: Proceedings of the international Conference on Formal ontology in information Systems. FOIS 2001, Ogunquit, Maine, USA, October 17 - 19, 2001, vol. 2001, pp. 34–46. ACM, New York (2001)
6. Guarino, N., Guizzardi, G.: In the Defense of Ontological Foundations for Conceptual Modeling. Invited Paper. Scandinavian Journal of Information Systems 18(1) (2006) ISSN 0905-0167
7. Guizzardi, G.: On Ontology, ontologies, Conceptualizations, Modeling Languages, and (Meta) Models. In: Vasilecas, O., Edler, J., Caplinskas, A. (eds.) Frontiers in Artificial Intelligence and Applications, Databases and Information Systems IV. IOS Press, Amsterdam (2007a) ISBN 978-1-58603-640-8
8. Guizzardi, G.: Ontological Foundations for Structural Conceptual Models, Ph.D. Thesis, University of Twente, The Netherlands (2005)
9. Guizzardi, G., Wagner, G., Guarino, N., van Sinderen, M.: An Ontologically Well-Founded Profile for UML Conceptual Models. In: Persson, A., Stirna, J. (eds.) CAiSE 2004. LNCS, vol. 3084, pp. 112–126. Springer, Heidelberg (2004)
10. Nunes, B.G., Guizzardi, G., Filho, J.G.P.: An Electrocardiogram (ECG) Domain Ontology. In: Proceedings of the Second Brazilian Workshop on Ontologies and Metamodels for Software and Data Engineering (WOMSDE 2007), 22nd Brazilian Symposium on Databases (SBBD)/21st Brazilian Symposium on Software Engineering (SBES). João Pessoa, Brazil (2007)
11. Oliveira, F., Antunes, J., Guizzardi, R.S.S.: Towards a Collaboration Ontology. In: Proceedings of the Second Brazilian Workshop on Ontologies and Metamodels for Software and Data Engineering (WOMSDE'07), 22nd Brazilian Symposium on Databases (SBBD)/21st Brazilian Symposium on Software Engineering (SBES), João Pessoa, Brazil (2007)
12. Object Management Group (OMG): Meta Object Facility MOF Core Specification, v2.0, Doc.# ptc/06-01-01 (2006)
13. Object Management Group (OMG): Object Constraint Language, v2.0, Doc.# ptc/06-05-01 (2006)
14. Benevides, A.B.: A Model-Based Tool for Well-Founded Conceptual Modeling (in portuguese), Computer Engineering Monograph, Computer Science Department, Federal University of Espirito Santo (2007)
15. Eclipse, `http://www.eclipse.org`
16. Dean, D., Gerber, A., Moore, B., Vanderheyden, P., Wagenknecht, G.: Eclipse Development using the Graphical Editing Framework and the Eclipse Modeling Framework, IBM Redbooks (2004)

17. Eclipse Modeling Framework Project (EMF), http://www.eclipse.org/modeling/emf
18. Model Development Tools (MDT), http://www.eclipse.org/modeling/mdt
19. Graphical Modeling Framework (GMF), http://www.eclipse.org/modeling/gmf
20. OntoClean, http://www.ontoclean.org
21. SUMO, http://www.ontologyportal.org
22. IEEE, http://suo.ieee.org
23. Protégé, http://protege.stanford.edu
24. AEON, http://ontoware.org/projects/aeon
25. Sandpiper Software, http://www.sandsoft.com
26. DOE, http://homepages.cwi.nl/~troncy/DOE
27. Sigma, http://sigmakee.sourceforge.net
28. Vissers, C., van Sinderen, M., Pires, L.F.: What Makes Industries Believe in Formal Methods. In: Proceedings of the 13th International Symposium on Protocol Specification, Testing, and Verification (PSTV XIII), pp. 3–26. Elsevier Science Publishers, Amsterdam (1993)
29. Guizzardi, G., Halpin, T.: Ontological Foundations for Conceptual Modeling. Applied Ontology 3(1-2), 91–110 (2008) ISSN 1570-5838

Concepts-Based Traceability: Using Experiments to Evaluate Traceability Techniques

Rodrigo Perozzo Noll and Marcelo Blois Ribeiro

Pontifical Catholic University of Rio Grande do Sul - PUCRS, Porto Alegre, Brazil
{Rodrigo.Noll,Marcelo.Blois}@pucrs.br

Abstract. Knowledge engineering brings direct benefits to software development through the cognitive mapping between user expectations and software solution, checking system consistency and requirements conformance. One of the potential benefits of knowledge representation could be the definition of a standard domain terminology to enforce artifacts traceability. This paper proposes a concepts-based approach to drive traceability by the integration of knowledge engineering activities into the Unified Process. This paper also presents an experiment and its replication to evaluate precision and effort variables from concepts-based traceability and conventional requirements-based traceability techniques.

Keywords: Traceability, Knowledge Engineering, Experimental Software Engineering.

1 Introduction

Several software development approaches use semi-formal or informal artifacts to capture different perspectives of a system under construction. These artifacts are structured in models with different abstraction levels, helping developers to think about software architecture and behavior. The use of knowledge resources is essential to software development, interchanging several aspects of information systems into a reasonable and understandable perspective.

It is natural for humans to first understand system concepts and behaviors under a business perspective before exploring the analytical and design perspectives. Software development is a process of abstraction and decomposition of conceptual entities into formal, computable ones. The work products produced during software development are derived from these concepts, suggesting a straightforward thinking for traceability.

The links between software work products and business requests are essential to practical software engineering. Although functional, the common approaches suffer from several limitations, including static and inflexible links, restrictive document formats, legacy artifacts and a high granularity for software matching.

Knowledge representation formalisms can be used to explicitly specify the domain model that represents certain application. This formalism can support the concepts mapping into the architectural elements, driving consistency of linking's dynamic aspects. In this context, the concepts-based traceability is suggested to provide a more specific granularity and wide flexibility than traditional traceability approaches do.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 539–550, 2009.
© Springer-Verlag Berlin Heidelberg 2009

To explore some aspects of concepts-based traceability, this paper evaluates the efficiency and usability of the proposal in a specific context. A first controlled experiment was performed to evaluate the concepts-based and the traditional requirements-based traceability techniques. As preliminary results, some conclusions about the proposal were drawn providing inputs for the experiment replication. The replication varied the manner in which the experiment ran but didn't vary the research hypothesis, according to the guidelines presented in [1]. After both controlled experiments, some lessons learned about the proposal were presented in order to contribute to the state-of-the-art in traceability and empirical studies for software engineering.

The paper is structured as follow: Section 2 presents a proposal to integrate knowledge engineering activities into the Unified Process (UP). Section 3 presents the concepts-based traceability, the tool developed to assist the process of indexing and retrieving artifacts and the related work. Section 4 presents the first experiment while Section 5 presents its replication. The lessons learned from the experiments are presented in Section 6 and Section 7 concludes the paper presenting the final remarks.

2 Knowledge Engineering and UP

The software development process states all activities, roles and work products required to build an information system. The UP is an example of a well-accepted software development process. During the earlier phases of the UP, the team is concerned to understand business process and to model the domain rather than construct the software. The focus is to discover, clarify and collect the problem oriented by knowledge. Business analysts and domain experts conduct the activities to model the universe of discourse, developing among others a work product called Domain Model, which describes the knowledge and its concepts using UML class diagram syntax.

There are several formalisms for knowledge representation. Ontology, for example, is a formalism used to specify a vocabulary of terms and relations with which the domain knowledge can be shared and reused. There are different kinds of ontology according to their level of generality, like top-level ontologies, domain ontologies and application ontologies [2]. For specific and contextualized information systems, application ontologies are used to represent the roles played by domain entities while performing some activity.

It is important to notice the similarity between ontologies and the Domain Model: both specify concepts and their relationships in the context of a domain of discourse and both are used to understand and share this conceptualization between the stakeholders. The process to capture the domain is directly related to the individual capability to gather and formalize it, requiring a high level of human expertise in knowledge representation techniques.

A preliminary and high-level ontology can be extracted from the Domain Model and its integrity could be maintained throughout the software life cycle by successive refinements. This refinement should identify new concepts, organize the taxonomy and define the semantic rules that describe the domain. The use of ontologies in the early development phases enables the concepts-based traceability between the domain

concepts and the architectural software elements derived from these concepts. The development of an ontology requires a systematic approach the same way as software development requires.

2.1 Ontological Engineering

The literature provides several approaches for systematic ontology development. It was necessary to evaluate the following proposals extracting the main aspects required to build an ontology: [3], [4], [5] and [6]. These proposals are based on an iterative process that includes the following steps: (1) Definition of the ontology scope; (2) Acquisition of the domain concepts (conceptual model); (3) Formalization of the concepts (formal model); (4) Integration to existing ontologies; (5) Definition of the axioms; (6) Validation.

Most of the efforts required to develop an ontology are related to domain comprehension and conceptualization. The same efforts are required during the early phases of the UP. In this situation, the effort required to develop an ontology could be reduced if the related UP artifacts were used as input. Using it to model the application knowledge, the domain concepts could be linked to all the others work products produced during the software life cycle.

To foster concepts-based traceability and to support the development of knowledge-based software engineering, a proposal to integrate ontology into the UP is presented. This proposal suggests a new discipline that gathers the tacit and explicit knowledge about the application domain.

2.2 Knowledge Engineering Discipline

The proposal to integrate ontologies into the UP does not modify any of the existing disciplines. Also, this paper does not intend to present a new approach for knowledge or ontology engineering but refers to the existing ones and how their activities could be related to the UP. The proposal creates *Knowledge Engineering* as a new discipline responsible for knowledge management and for the semantic interface between software work products.

The main steps presented in Section 2.1 are included to define the Knowledge Engineering discipline. Also, some of the UP work products are used as input. The discipline is structured over three major activities: Design, Maintenance and Validation. The ontology is designed in the early phase, including steps 1, 2 and 3. The design continues until the Domain Model is completed. Once it is completed, the ontology can be automatically extracted from the UML class diagram and maintenance activity can start, executing steps 4 and 5. Every modification in the ontology must be validated as stated at step 6. The following sections will detail each activity presented in the discipline's workflow.

Design. The design activity defines the ontology scope and its primary concepts. This activity also supports taxonomic and non-taxonomic properties definition. The design activities are executed by a new role called *Knowledge Engineer*. The Knowledge Engineer activities include the refinement of the Domain Model based on the taxonomic properties and non-taxonomic object and data type properties. During this activity, no complex restriction rule is defined as standard UML class diagrams do not

support it. After finalizing the Domain Model, the next activity is to extract a preliminary version of the ontology, generating a new work-product called *Ontology*.

The ontology is extracted using the "Mapping UML to OWL" proposed by OMG Ontology Definition Metamodel (ODM) and summarized in Table 1. The language used to manipulate the ontology is OWL since it is officially recommended by W3C.

Table 1. Summary Mapping between UML and OWL

UML	OWL
Class	OWL Class
Attributes	Datatype Property Domain: Parent class Range: XSD defined by XML Schema.
Associations	Object Property Domain: Source association class. Range: Target association class. Restrictions: 1. If the initial association and the final association are navigable, so the property is symmetric. 2. If the multiplicity of both associations is the same, generate a new cardinality restriction. If not, create a maximum and minimum cardinality restriction.
Generalization	Defines the taxonomy between classes

Maintenance. The Design activity does not produce a complete version of the ontology because it does not include complex semantic rules. The ontology maintenance could be done using the approach suggested by [6]: first analyzing the taxonomic organization and refining the ontology when needed; after that, inspecting non-taxonomic relations that could be refined. This approach will manage the domain concepts while creating all the system work products.

This paper will not explore how to extract complex rules from business perspective. For this purpose, Section 2.1 presents several approaches that address this subject.

For the purpose of the concepts-based traceability, the main goal of Maintenance activity is the mapping between software specification and domain concepts. During the development process, every single element from software work products could be mapped into the ontology concepts. Additionally, these links will represent the tacit knowledge acquired and appropriated by the persons involved that often exists only in their mind. Later will be present how these links are managed and retrieved.

Like the previous activity, the Knowledge Engineer is responsible for the Maintenance activities, receiving as input the Ontology, the Requirement Specification and the Analysis and Design Models. The Knowledge Engineer will update the ontology throughout the whole development cycle, adding traceability links to the model elements and refining the concepts, properties and semantic rules.

Validation. The Validation activity provides the regular evaluation of the ontology integrity. This activity also suggests existing approaches to validate the knowledge modeled. It is suggested to use the approaches presented by [4] and by [6], validating the consistency of the logic model using an inductive and pragmatic approach. The validation grants a consistent and coherent representation of the application domain. This process also proposes three validation types for each part of the ontology

development: unit test, integration test and acceptance test. The Knowledge Engineer should first execute the first two and software agents or other applications that uses ontologies should execute the last one.

3 Concepts-Based Traceability

The knowledge formally specified could be used for several tasks, like software consistence checking, artifacts sharing and inter-operation and communication between the stakeholders through a common non-technical language.

The concepts-based traceability is one of the possible benefits related to the Knowledge Engineering discipline. This traceability brings two main benefits: lower granularity degree of the traced elements and the cognitive links extracted by inference engines.

The linking functionality is automatically generated into the ontology extracted from the UML Domain Model. The ONTrace resource is an OWL class related to an object property (ontraceRecover) that implements the mapping between UML elements (such as use cases, classes in a class diagram, or any other element) and the knowledge resources (OWL classes or properties). For every single traceability link, an instance of ONTrace resource is created or updated, mapping one knowledge resource to certain UML element.

To present an example about the concepts-based traceability, let's consider an e-commerce scenario. Suppose that the use case "*Maintain Client*" is related to the ontology concept "*Client*". Another use case "*Buy Product*" is related to the ontology concepts "*Client*" and "*Product*". So, it is possible to say that "*Maintain Client*" and "*Buy Product*" are explicit related to the "*Client*" concept (a direct relation link).

In the same example, suppose that there is a use case called "*Maintain Employee*" that is related to the concept "*Employee*". Also, consider that the object property "*getInformation*" relates the concept "*Employee*" to the "*Client*". In this scenario, it is possible to infer an implicit relationship between the previous "*Maintain Client*" and "*Maintain Employee*" because the concepts that relate both are associated by the "*getInformation*" property. It is important to notice that the use of complex semantic rules into an inference engine allows the retrieval of scenarios much less obvious than the example presented.

3.1 ONTrace: A Tool for Concepts-Based Traceability

The ONTrace tool was developed extending ArgoUML [7]. The tool contains four basic functionalities. The first one is the automatic ontology extraction from the UML class diagram referred in the Design activity. To support the Maintenance activity, the tool also exports and imports the application ontology. This functionality is required for the Knowledge Engineer adds restriction rules into the ontology. In this step, the use of some ontology-editing tool like Protègè [8] is suggested.

Based on the ontology resources, a panel at the bottom of the tool is populated with OWL classes, object properties and data type properties. The third functionality is the ability to link each UML elements to the ontology resources, only checking a box besides the resource name. The last functionality retrieves the traced links, enabling users to perform queries over the ontology using an inference engine.

The tool ONTrace+ArgoUML was used to empirically evaluate the concepts-based traceability proposal.

3.2 Related Work

The purpose of this paper is not to present a new approach to guide ontology development or to specify semantic rules, but to introduce the concepts-based traceability as one of the possible benefits of the knowledge engineering integration into the software development. There are several approaches for ontology construction like presented in Section 2.1. Essentially, these approaches provide the basis for the Knowledge Engineering discipline definition.

Regarding the ontologies integration into the software development, there are several related works like [2] and [9]. Those proposals intend to develop information systems based on the ontology definition. Our proposal does not suggest changes to the existing development paradigm, but new activities for knowledge management during development life cycle in order to support concepts-based traceability.

The concepts-based approach reduces the link granularity fostering a better matching than traditional requirements-based traceability techniques do. Traditional traceability approach has limitations such as static and inflexible links from a single requirement to several work products, retrieving only explicitly stored links.

Using concepts for traceability seems to be more precise than traditional approaches as one requirement gathers different concepts. The precision using the concepts approach may require additional effort to link the elements since more links must be generated. To evaluate the specific aspects of precision and effort measures between traceability approaches, some experiments were conducted.

4 The Experiment

The experiment requires a formal process to conduct and analyze a subject in order to control the variables that could influence it. The experiments definition and evaluation were done using the guidelines presented at [1] and [10].

4.1 Definition

The experiment definition was structured using the GQM approach [11]. The goal of this study was to analyze the concepts-based (μcon) and the requirements-based (μreq) traceability approaches during the UP, for the purpose of characterizing the time required and the elements retrieved by each approach, with respect to their effort and precision, from the point of view of a software architect and in the context of the maintenance of a sports academy management system. From this goal, the following questions and metrics were derived:

1. Is the effort required to define the traceability links using μcon the same as μreq?
   Metric: difference between the final and initial time (ΔT) in minutes.
2. Is the precision of the retrieved artifacts using μcon the same as using μreq?
   Metric: defined by Jaccard's coefficient and illustrated by the equation below.

$$P = \frac{amountOf\left(R \cap A\right)}{amountOf\left(R \cup A\right)} \quad (1)$$

Where **R** stands for the set of artifacts recovered using some approach and **A** for the set of actual related artifacts defined by software inspection.

4.2 Planning and Operation

The traceability experiment was conducted in a university using a controlled, in-vitro and off-line environment. The subjects were twelve undergraduate and graduate students with similar industry experience. The population sampling was non-probabilistic and it was conveniently chosen by quote.

The previous questions derived two hypotheses:

1. The effort needed to identify traceability links is the same for both approaches.

$$\mathbf{H_0}: \Delta T_{\mu con} = \Delta T_{\mu req} \quad \mathbf{H_1}: \Delta T_{\mu con} > \Delta T_{\mu req} \quad \mathbf{H_2}: \Delta T_{\mu req} > \Delta T_{\mu con}$$

2. Using the same query, both approaches retrieve the same set of artifacts.

$$\mathbf{H_0}: P_{\mu con} = P_{\mu req} \quad \mathbf{H_1}: P_{\mu con} > P_{\mu req} \quad \mathbf{H_2}: P_{\mu req} > P_{\mu con}$$

The independent variables were μcon and μreq approaches and the dependent variables were effort and precision metrics. The design was one factor with two treatments. The instrumentation included the following work products related to a small sport academy management system: one design class diagram, nine use case descriptions and the domain model with nine concepts discriminated. The traceability links were created using ONTrace+ArgoUML tool for μcon and MS Excel for μreq. Forms fulfilled by the subjects were used to collect measurement data.

During the operation, specific training and guidelines were used to motivate the subjects while presenting each technique, avoiding the confounding factor. The μreq received nine use case descriptions and one design class diagram. The subjects were requested to check the boxes indicating what classes are related to each use case. The μcon linked nine ontology concepts into the same design classes, by checking a box. For both approaches, the effort was captured from this linking activity. To acquire the precision variable, some design classes were requested to be traced, listing all the others related elements. The queried classes were the same for both approaches and the time was not considered.

4.3 Analysis and Interpretation

For data analysis, the minor degree of freedom (p-value) adopted for all tests was 5%. The experiment execution produced the data set presented in Table 2.

Table 2. Experiment execution data set

Approach	Subject	Effort (min)	Precision (P)	Approach	Subject	Effort (min)	Precision (P)
µcon	C01	4.00	1.00	µreq	R01	20.00	0.51
µcon	C02	5.00	1.00	µreq	R02	17.00	0.34
µcon	C03	3.00	1.00	µreq	R03	22.00	0.51
µcon	C04	8.00	0.69	µreq	R04	20.00	0.51
µcon	C05	4.00	1.00	µreq	R05	30.00	0.51
µcon	C06	4.00	1.00	µreq	R06	22.00	0.53

First Hypothesis: Effort. The initial analysis focused on evaluating data distribution and identifying possible outliers. The Box Plots was generated and, even finding R05 as an outlier, no subject was removed from the hypothesis testing. In a data set of only 6 points, leaving out any data point should not be done based only on box plots analysis and besides no further evidence was found to justify removing this outlier.

Two hypotheses were defined to evaluate data distribution: $\mathbf{H_0}$: data has a normal distribution; $\mathbf{H_1}$: data has a non-normal distribution. Shapiro-Wilk test was used to test the previous hypotheses as this experiment has less than 50 subjects. The result for µcon was 0.055 and for µreq was 0.171, which are greater than 5% for both approaches. In this case, it was not possible to prove that the dataset distribution is not normal. From this point on it is assumed that the sample is normally distributed

To evaluate homoscedasticity, the following hypotheses were defined: $\mathbf{H_0}$: Sample has the same variance; $\mathbf{H_1}$: Sample has not the same variance. The result from Levene's Test for the same variance was 0.262, which is greater than the p-value assumed. Thus, H_0 cannot be rejected which implies that the parametric T-Test can be applied for the hypotheses testing defined in Section 4.2. The T-Test results for the criterion that rejects H_0 by H_1 are presented in Table 3.

Table 3. T-Test for the effort variable evaluating H_1

Same Variances	T	Degr.Freedom	Significance(two-tailed)
Yes	-8.878	10	0.000

Comparing the value from the calculated T (– 8.878) to the tabulated value (1.812), H_0 cannot be rejected by H_1 because T value is lower than the tabulated value. The second T-Test was applied to evaluate H_0 by H_2 and its results are presented below.

Table 4. T-Test for the effort variable evaluating H_2

Same Variances	T	Degr.Freedom	Significance(two-tailed)
Yes	8.878	10	0.000

Comparing the value from the calculated T (8.878) to the tabulated value (1.812), H_0 can be rejected and H_1 can be accepted because T value is greater than the tabulated value. The presented analysis concluded that exists in this specific

experiment a statistical difference between the efforts related to the definition of traceability links. The effort related to the requirements-based traceability is greater than the concepts-based one.

Second Hypothesis: Precision. Using the same approach as before, even identifying moderate outliers as C4, R2 and R6 by Box Plots analysis, no point was removed from the data set. Also, the same two hypotheses previously defined to evaluate data distribution were used and the results from Shapiro-Wilk test for μcon was 0.584 and for μreq was 0.496. The null-hypothesis was rejected because the significance is lower than the p-value adopted. As consequence, the non-parametric Mann-Whitney test was executed to evaluate the following hypotheses: H_0: samples are from the same distribution; H_1: samples are not from the same distribution. The asymptotic significance was 0.002, which is lower than the p-value assumed, so H_0 could be rejected. Consequently, both samples are from distinct distribution (there is a statistical mean difference between μreq and μcon). The Mann-Whitney test can reject H_0 but cannot evaluate H_1 and H_2, checking which technique has better precision factor. This question was addressed by comparing the arithmetical means, that for μcon was 0.9483 and for μreq was0.4850. Although this result is questionable in terms of statistical significance, the simple comparison of arithmetical means indicates that the precision related to μcon is greater than μreq.

5 Experiment Replication

The purpose of an experiment replication is to define the basis for meta-analysis and to confirm that the previous results were not unsupported in some aspect. The replication was executed almost one year after the first instance was completed, using different subjects and instrumentation, but focusing on the same objectives and hypotheses. The type of replication used was varying the manner in which the experiment has run but not the hypotheses [1].

5.1 Definition, Planning and Operation

The experiment replication used the same definition as the first instance. The context selection was a university and the subjects were eighteen graduate students with similar industry background and knowledge about software development. The hypothesis formulation, variable selection, environment, reality and design defined in the first instance were maintained.

During the replication, the instrumentation has changed from the design class diagram to three sequence diagrams. The use case descriptions and the domain model didn't change. For the execution, new treatments, guidelines and training were developed.

The operation consisted of linking use cases or domain concepts from the sequence diagram into methods. The precision and effort variables were gathered in the same way then the first execution.

5.2 Analysis and Interpretation

The experiment execution produced the data presented in Table 5.

Table 5. Experiment execution data set

Approach	Subject	Effort (min)	Precision (P)	Approach	Subject	Effort (min)	Precision (P)
µreq	R01	25	0.5092	µcon	C01	16	0.7024
µreq	R02	22	0.6520	µcon	C02	21	0.7202
µreq	R03	16	0.6520	µcon	C03	18	0.3000
µreq	R04	31	0.1282	µcon	C04	22	0.8346
µreq	R05	6	0.5238	µcon	C05	10	0.8222
µreq	R06	34	0.4286	µcon	C06	20	0.3939
µreq	R07	10	0.5092	µcon	C07	27	0.7598
µreq	R08	20	0.2857	µcon	C08	23	0.8839
µreq	R10	15	0.6520				

First Hypothesis: Effort. Despite having moderate outliers in the data set, none of them were removed. To evaluate data distribution, the following hypotheses were defined: H_0: the data has a normal distribution; H_1: the data has a non-normal distribution. The result from Shapiro-Wilk for µcon was 0.868 and for µreq was 0.973, which is greater than 5% in both approaches, so it is not possible to prove that the dataset distribution is not normal. For homoscedasticity, the following hypotheses were defined: H_0: Sample has the same variance; H_1: Sample has not the same variance. Using Levene's Test, the significance degree for same variances was 0.153, which is greater than the p-value assumed, so H_0 cannot be rejected. As result, it was used the parametric T-Test to evaluate the hypotheses defined at Section 4.2 and its results are presented in Table 6.

Table 6. T-Test for the effort variable evaluating H_1

Same Variances	T	Degr.Freedom	Significance(two-tailed)
Yes	-0.164	16	0.872

Comparing the value from the calculated T (-0.164) to the tabulated value (1.746), H_0 cannot be rejected by H_1 because T value is lower than the tabulated value. The second T-Test was applied to evaluate H_0 by H_2 and its results are presented below.

Table 7. T-Test for the effort variable evaluating H_2

Same Variances	T	Degr.Freedom	Significance(two-tailed)
Yes	0.164	16	0.872

Comparing the value from the calculated T (0.164) to the tabulated value (1.746), H_0 cannot be rejected by H_2 because the T value is lower than the tabulated value. The arithmetical mean was extracted and the result for µcon was 19.625 and for µreq was 20.777. Despite the change from the design class diagram, which is closer to the

domain model, to the sequence diagram, which is closer to the use case behavior scenarios, the effort required to μreq is a little greater than the required to μcon.

Second Hypothesis: Precision. Box plot analysis indicated moderated outliers, but none of them were removed from the data set. The same two hypotheses previously defined to evaluate data distribution were used and the result from Shapiro-Wilk test for μcon was 0.061 and for μreq was 0.080. The null-hypothesis could not be rejected because the significance is greater than the p-value adopted. To evaluate homoscedasticity, the previous two hypotheses were used and the results for Levene's test for the same variances was 0.444, indicating that H_0 cannot be rejected. The parametric T-Test was executed and the result for the calculated T was 2.132. Comparing the calculated value to the tabulated value (1.746), H_0 can be rejected by H_1 because T value is greater than the tabulated value. This statistical test could not be executed the first time the experiment was executed, raising questions about the results statistical confidence. During the replication, it was possible to prove that, for this specific context, the precision related to the concepts-based traceability is greater than requirements-based traceability.

6 Lessons Learned and Final Remarks

Usually, the cognitive and implicit mapping between what is being developed during software life cycle is managed by the development staff. These links are hard to maintain, mainly when teams are working apart. Unless there is a managed communication chain between the team members, it is impossible to keep track of all relevant knowledge and its evolution during the software development projects.

In order to include knowledge engineering activities into the traditional Unified Process, this paper highlighted some well-know approaches to build an ontology. Using these approaches as basis, the Knowledge Engineering discipline was proposed to be included into the UP. It is known that knowledge engineering is not a new discipline and it is widely used in approaches like Product Line, but what is discussed here is how to take advantage of existing UP work products to build an ontology, enabling the traceability proposal. This paper also presented a generic structure to create and retrieve traceability links and a tool that automates this approach.

The experiments presented in this paper are characterized as primary studies to quantitatively analyze two competing approaches for traceability. These in vitro experiments were executed to evaluate relevant hypotheses in a very specific context, setting up a knowledge baseline that should be iteratively increased. This expansion must include in vivo empirical studies relating real products and processes.

In general basis, the experiment reveals that the concepts-based approach requires less effort and retrieves more precise elements than requirements-based traceability does. Although the extracted results cannot be generalized to the entire development process, the proposal applicability provides some evidences. Most of the efforts required to the requirements-based traceability consist of identifying what software elements matches to use case descriptions. For this purpose, the requirements must be clearly understood to accurate trace it to software elements. As the domain concepts are more specific and are defined in a non-technical language, the tracing tends to be easier for the subjects understand and execute. This evidence gets clear during this both empirical studies while analyzing effort variable.

Some evidences about the precision variable could also be extracted: as one use case can index distinct model elements, it also retrieves false positive elements during impact analysis. This can be explained by the specific granularity of concepts-based approach if compared to requirements-based ones, that encapsulates several domain concepts.

Another evidence is related to the instrument used: the first experiment explored the structural perspective of a design class diagram, which is closer to the domain concepts structural perspective than the use case behavioral perspective. The experiment replication removed this tendency exploring the behavioral perspective of a sequence diagram, closer to the use case description. Even then, the results for the effort and precision variable did not turn favorable to requirements-based traceability as tendency suggests.

Despite the limitations presented by this empirical and controlled study, including a small population, the proposed traceability approach provides some evidences encouraging further research. The future work will include replications in vivo to improve the existing baseline and to increase scientific knowledge and experimental evidences about this object of study.

Acknowledgement. Study developed by the Research Group in Intelligent Systems Engineering Group of the PDTI, financed by Dell Computers of Brazil Ltd. with resources of Law 8.248/91.

References

1. Basili, V.R., Lanubile, F.: Building knowledge through families of experiments. IEEE Transactions on Software Engineering 25(4) (1999)
2. Guarino, N.: Formal Ontology in Information Systems. In: Proceedings of FOIS 1998, pp. 3–15. IOS Press, Amsterdam (1998)
3. Gruninger, M., Fox, M.S.: Methodology for the Design and Evaluation of Ontologies. In: Workshop on Basic Ontological Issues in Knowledge Sharing, pp. 234–241 (1995)
4. Fernández, M., Gómez-Pérez, A., Juristo, N.: Methontology: From Ontological Art towards Ontological Engineering. In: Proc. AAAI 1997, pp. 33–40 (1997)
5. Noy, N.F., McGuinness, D.L.: Ontology Development: A Guide to Creating Your First Ontology. Technical Report KSL-01-05, Stanford Knowledge Systems Laboratory and Stanford Medical Informatics (2001)
6. Sure, Y., Studer, R.: On-To-Knowledge Methodology - Final Version. On-To-Knowledge EU IST-1999-10132 Project Deliv. D18, University of Karlsruhe (2002)
7. ArgoUML - A UML design tool, `http://argouml.tigris.org`
8. The Protègè Ontology Editor, `http://protege.stanford.edu`
9. Falbo, R.A., Ruy, M.R.D.: Using Ontologies to Add Semantics to a Software Engineering Environment. In: Proc. of 17th International Conference on Software Engineering and Knowledge Engineering, pp. 151–156 (2005)
10. Wohlin, et al.: Experimentation in software engineering: an introduction. Kluwer Academic Publishers, USA (2000)
11. Basili, V.R., Caldiera, G., Rombach, H.D.: The Goal Question Metric Approach; Encyclopedia of Software Engineering. Wiley-Interscience, New York (1994)

A Service-Oriented Framework for Component-Based Software Development: An i* Driven Approach

Yves Wautelet, Youssef Achbany, Sodany Kiv, and Manuel Kolp

Info. Syst. Research Unit (ISYS), University of Louvain, Belgium
{yves.wautelet,youssef.achbany,sodany.kiv,
manuel.kolp}@ucLouvain.be

Abstract. Optimizing is a fundamental concept in our modern mature economy. Software development also follows this trend and, as a consequence, new techniques are appearing over the years. Among those we find services oriented computing and component based development. The first gives the adequate structure and flexibility required in the development of large industrial software developments, the second allows recycling of generically developed code. This paper is at the borders of these paradigms and constitutes an attempt to integrate components into service-oriented modelling. Indeed, when developing huge multi-actor application packages, solutions to specific problems should be custom developed while others can be found in third party offers. FaMOS-C, the framework proposed in this paper, allows modelling such problems and directly integrates a selection process among different components based on their performance in functional and non-functional aspects. The framework is firstly depicted and then evaluated on a case study in supply chain management.

Keywords: Requirements Engineering, Service-Oriented Modeling, i*, Multi-Agent Systems.

1 Introduction

The development of large industrial application packages involving different actors played by a series of collaborating or competing companies leads to huge developments which are never homogenous. Indeed, if different market actors decide to join forces for developing a collaborative application package in order to, for example, managing a whole supply chain as in the TransLogisTIC project [1]; analysts need advanced tools to describe organisations or user requirements to make the best possible design architecture. However, such developments are never homogenous since a series of components developed and provided by third party (proprietary or open source software or even developments of the actors themselves) and dedicated to specific tasks are already existing and interesting to adopt. Huge software developments should consider those in its analysis and design process especially when facing a collaborative platform linking different actors developed on the basis of services technology. That is why, in this paper we propose FaMOS-C, a service-oriented FrAMework fOr maS modelling and Component selection. This framework extends the traditional FaMOS framework (presented in [2]) to integrate components identification and selection into the development process.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 551–563, 2009.
© Springer-Verlag Berlin Heidelberg 2009

A supply chain is the set of all actors and relations between them which participate in the process of delivering value to a customer, as a product or service. It includes all the processes from raw materials to delivery and is viewed as a network of information, materials and financial flows. In this paper we particularly focus on the application of a service-oriented methodology for - possibly component-based - software development on outbound logistics (i.e. the process related to the movement and storage of products from the end of the production line to the end user) with a strong focus on transportation. The application package development is focused on the integration of different kinds of services allowing collaboration on a global basis. Among those services a series of can be fulfilled using third party components. The idea underlying the developed collaborative application package is to offer the most advanced services to optimize the chain on the most global basis with different quality of service.

The paper is structured as follows. Section 2 discusses the problem statement and main contributions of the paper. Section 3 presents a conceptual framework for component-selection in service-oriented MAS developments. Section 4 overviews a case study, the development of a collaborative supply chain application focusing on outbound logistics. It presents the service-oriented analysis combining custom developments as well as components selection. A component selection process is presented in the section. Section 5 is the related work. Finally, Section 6 concludes the paper.

2 Problem Statement

This section introduces component-based software development as well as the services approach. Supply chain management is also overviewed and the application of service-oriented development with COTS is depicted.

2.1 Component-Based Software Development

Component-based Software Development (CBSD) is based on the idea to develop software systems by selecting appropriate commercial off-the-shelf (COTS) components and then assembling them with well-defined software architecture [3].

Following [4], a component has three main characteristics:

- it constitutes an independent and replaceable part of a system that fulfils a clear function;
- it works in the context of a well-defined architecture;
- it communicates with other components by its interfaces.

The life cycle and software engineering model of CBSD is much different from that of the traditional ones, with more effort put into requirements, test and integration, and less in design and code. Most models encountered in literature consider the activities of identification, selection, integration and adaptation, as part of the development process to construct systems based on COTS components.

The CBSD approach has raised a tremendous amount of interests both in the research community and in the software industry. Following [4,5], the main advantages of the CBSD approach are:

- better management of the application complexity;
- decrease of the development cost and time;
- increased flexibility;
- increased quality (mature solutions from which correctness has been established in earlier projects).

Although the CBSD promises significant benefits, there are some technical problems that limit their use. Among those problems, we find:

- how the business requirements must be captured and refined, based on a process that leads to the development of a component based system;
- how the components must be put together and deployed using the latest technologies;
- how to select the components that are the closest to the developers' needs.

FaMOS-C, the development framework proposed in this paper partially addresses these issues.

2.2 Service-Oriented Development

Service-oriented computing is becoming increasingly popular in the development of large and flexible software. Such approach fits our goal to dispose of a framework for developing large industrial software application packages combining self-developed modules with third party components. Moreover, services are used as fundamentals to drive the software process. They are distinguished and analysis early onto the project (through the FaMOS framework, see [2] and this paper for the refinement using components) then converted into service centers at design stage for an agent oriented implementation (this perspective is not developed into this paper but can be found in [2]). More precisely:

- The organisation is firstly described as a set of organisational and component services in the Strategic Services Diagram;
- Services are split in a series of goals and tasks depicted in the Strategic Dependency and Rationale Diagrams;
- Component services' related goals depicted in the SRD are then analysed using a NFR goal graph for adoption of third party components into the application package. Goals are transformed into operationalizations and a means-end analysis is performed;
- Organisational services are designed as services centers in the architectural design discipline (this aspect is not overviewed in this paper).

2.3 The Services Approach in Supply Chain Management

In the context of the development of a multi-actor collaborative application package we used, in this paper, a service-oriented approach to develop a MAS. This has been done following the ProDAOSS methodology (this methodology is described in [6,7]).

In our approach, component-based software engineering is directly considered into the services analysis. Components are a specialization of the Strategic Services Diagram's actors providing services to other actors so that we dispose of an integrated view of both services provided by components and other ones. The services approach in the context supply chains is of particular interest because:

- Supply chain software and data collection will, in the future, take the form of a utility accessible on demand;
- SOA could have a profound effect on the way supply chain software is designed, sold and implemented in the future. Common workflow based developments progressively leave place to generic customizable systems. In that perspective, SOA is designed to support rather than dictate business processes. This trend is for example also interesting in the ERP systems development perspective;
- Software developments' integration is from strategic importance in nowadays business context, SOA with its inherent flexibility ease such a process.

2.4 The Services Approach in Supply Chain Management

Paper's main contribution is the modelling framework for service software development. The later is:

- Heterogeneous: combining organizational services with component services;
- Offers multiple-views:
  - The SSD offers the most aggregate static view for adequate understanding of the software application package to develop;
  - The SRD offers a static view of the involved actors including the component actors and its depending goals;
  - The NFR goal graph for analysis of the functional and non-functional aspects of the adopted component.

The paper also applies the framework on a supply chain management case study to show its applicability and give the reader an idea of the developed framework.

3 FaMOS-C

This section further explains the FaMOS-C approach, an unified-framework for agent-oriented requirements engineering, and further refines the Strategic Services Diagram meta-model.

3.1 The Approach

As evoked earlier, the FaMOS-C framework presented here is directly inspired by FaMOS [2] but focuses on the component selection process. Indeed, the Strategic Services Diagram identifies organizational and user services but its meta-model will be refined into the next section to include the definition of component-services. The later are proposed by a third party vendor and are, in the framework, further documented

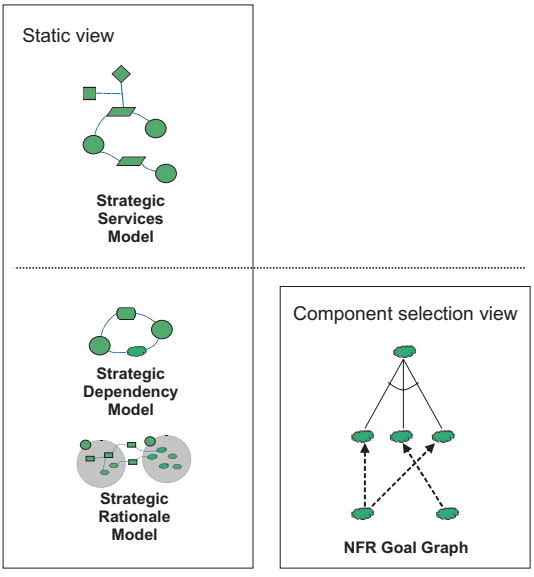


Fig. 1. The Framework

by a Strategic Rationale Diagram. In the later diagram, the SSD dependee agent of the documented service (which in the case of a component service is a component agent) is the dependee for a couple of goals involving one or more dependers. Those goals constitute the core functionalities the component has to offer, that is why they are distinguished as operationalizations in the NFR goal graph, refining the *generic* component service they refer to. More operationalizations are distinguished into the graph to cover the expected non-functional requirements. The NFR goal-graph is then used for the selection process of the best adapted component solution in the context of a particular software development. Figure 1 resumes the different views of the framework each having its own purpose. The component services' non-functional requirements (softgoals) and functional ones (goals) are evaluated on the basis of an NFR goal graph. Those are then compared to the measured (or estimated) performance of each vendor solution on those specific criteria. The best performing product when matching the expected non-functional requirements level with their available characteristics is considered as the most interesting candidate solution in the context of the project.

3.2 Conceptual Model

As exposed earlier, we define directly into the Strategic Services Diagram (first defined in [2]) the notion of *Component Service* as a refinement of service as well as the notion of *Third Party Provider* as a refinement of Actor (this can be visualized in the meta model of Figure 2, a complete description is available in [2]). Those are distinguished to highlight the particularities of those concepts compared to their root element. Into an SSD instance, the actors appear as circles while the services appear as rhombs, third

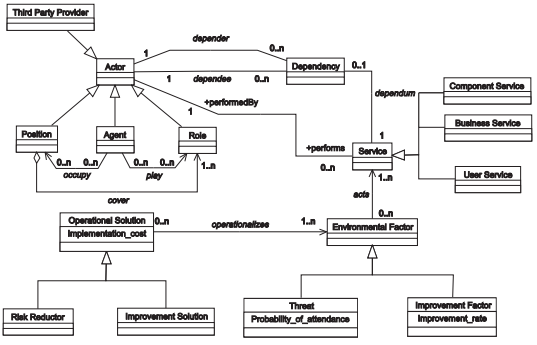


Fig. 2. Refined Strategic Services Diagram Meta-Model

party providers and components services are filled in mallow while others are in light blue on a black and white printing, the first two types of elements appear darker.

4 Case Study

This section overviews the application of the FaMOS-C framework onto a case study issued of supply chain management and more particularly outbound logistics. It overviews the different third party components that have to be introduced into the application packages, briefly describes the SSD and finally focuses on the adoption of a component solution for the *Track Transport service*.

4.1 Outbound Logistics

Outbound logistics is the process related to the movement and storage of products from the end of the production line to the end user. In the context of this paper we mostly focus on transportation. The actors of the supply chain play different roles in the outbound logistic flow. The producer will be a logistic client in its relationship with the raw material supplier, which will be consider as the shipper. The carrier will receive transportation orders from the shipper and will deliver goods to the client, while relying on the infrastructure holder and manager. In its relation with the intermediary wholesaler, the producer will then play the role of the shipper and the wholesaler will be the client. Figure 3 summarizes the material flows between the actors of the outbound logistics chain. The responsibilities corresponding to the different roles are:

- Shipper: has received an order from a client, and does a logistic request to a carrier for the delivery of that order.
- Carrier:
  - The Strategic Planner: decides on the services that are offered on the long term, on the use of infrastructure, on the logistic resources to hold and on the client's acceptation conditions.
  - The scheduler: orders transports to be realized, according to the strategic network and constraints, coordinates with the infrastructure manager and assign logistic requests to those transports such that the delivery requirements are met.

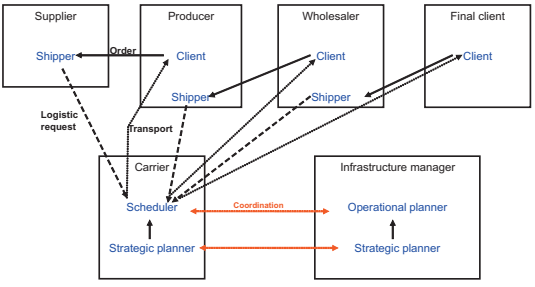


Fig. 3. Material flows in the outbound logistics chain

– Infrastructure Manager: holds the logistic infrastructure and coordinates with the carrier's scheduler to offer the network for the planned transports.

The idea underlying the software development is to favour these actors' collaboration. Indeed, collaborative decision will tend to avoid local equilibriums (at actor level) and wastes in the global supply chain optimisation, giving opportunities to achieve the greatest value that the chain can deliver at lowest cost (see [8,9]). The collaborative application package to develop is thus composed of a multitude of aspects including the development of applications and databases to allow the effective collaboration and the use of flexible third party components providing well identified services. This dual aspect is from primary interest as a case study of FaMOS-C, the framework we propose in this paper.

4.2 Third Party Components

Third party components that can fulfil an amount of the identified requirements in the context of the development of the collaborative software package are the following:

– The **Fleet Management System** (FMS) is computer software that enables people to accomplish a series of specific tasks in the management of a company's vehicle fleet. It can include vehicle telematics (tracking and diagnostics), driver management, fuel management, vehicle maintenance and so on;
– The **Warehouse Management System** (WMS) is a key part of the systems managing the supply chain. It aids in controlling the movement and storage of materials within a warehouse and processing the associated transactions, such as shipping, receiving, putaway and picking;
– The **Enterprise Resource Planning** (ERP) is an enterprise-wide information system designed to support most of the business system. It maintains in a single database the data needed for a variety of business functions such as Manufacturing, Supply Chain Management, Financials, Projects, Human Resources and Customer Relationship Management;
– The **Transportation Management Systems** (TMS) is computer software designed to manage transportation operations. It aids in determining the most efficient and most cost-effective way to execute the movement of product(s). The TMS will be further overviewed in the case study, it includes various functions such as:

- Planning and optimizing of terrestrial transport rounds;
- Transportation mode and carrier selection;
- Management of air and maritime transport;
- Real time vehicles tracking;
- Service quality control;
- Vehicle load and route optimization;
- Transport costs and scheme simulation;
- Shipment batching of orders;
- Cost control, key performance indicators (KPI) reporting and statistics.

4.3 Framework Application

This section presents the application of the framework for the selection of the TMS component for the development of a (much larger) collaborative software package. First of all, the whole package is presented, then the strategic rationale model allows enlightening the (functional) goals depending of the third party component and finally an NFR goal graph is drawn to analyze both functional and non-functional expected requirements. A table finally points to the best suited component for best fulfilling the service.

Application Services. The Strategic Services Diagram of Figure 4 introduces all the services of the application package. This view allows all project stakeholders to share a common aggregate view of the services including their dependency relationships. The darkest services are component services, we will, in the context of this case study focus one of them *Track Transport*, a service provided by the Transportation Management System, to the Shipper and the Carrier.

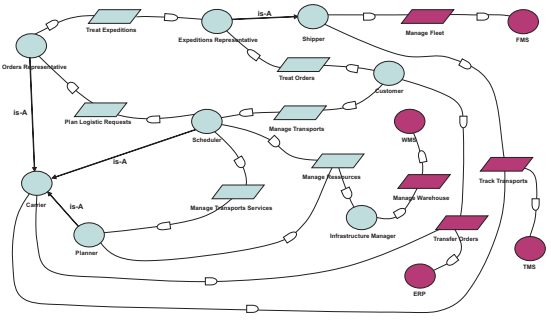


Fig. 4. Strategic Services Diagram for Outbound Logistics

A Focus on the Track Transports Service. The Strategic Rationale Diagram (SRD) of Figure 5 depicts the actors, goals, tasks and resources involved into the *Track Transports* service realization. This diagram offers a complete static view of the actors' dependencies for resources, goals and tasks realization. This view is from primary importance for documenting the goals and tasks expected for the service realization. Five actors are involved in the process of tracking transports with the carrier and the shipper dependent

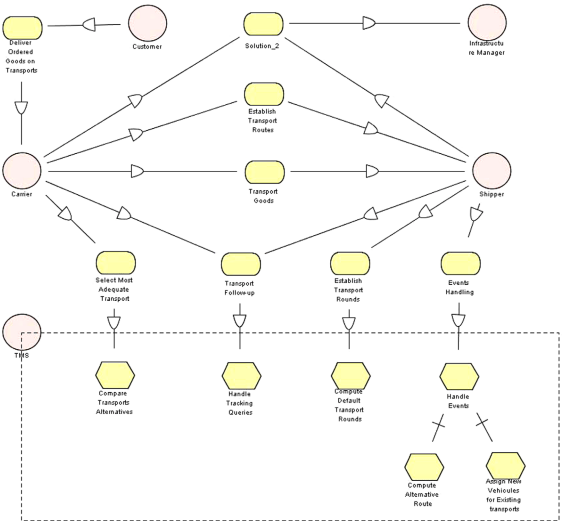


Fig. 5. Strategic Rationale Diagram for Track Transports

from the TMS. Those actors' goals depending of the TMS are *Select Most Adequate Transport*, *Transport Follow-up*, *Establish Transport Rounds* and *Events Handling*.

Finally, the NFR Goal graph offers a complementary view to the SRD presented here by showing the non-functional requirements importance in the context of a component selection. This is the subject of the next section.

4.4 Component Selection

The NFR goal graph of Figure 6 represents both the functional and non-functional aspects that the Track Transports service should fulfil. Non-functional requirements of this service include global optimization (of the supply chain), collaboration, flexibility and security. Note that the service itself is represented as a non functional requirement, see Track Transports in Figure 6 since it possesses high abstraction and regroups a series of operationalized goals. After posing these non-functional requirements as goals to satisfy, we attempt to refine them into sub-goals (operationalizing goals) and we address also the interdependencies among the goals as shown in Figure 6.

Figure 7 illustrates the evaluation of the TMS component from three different vendors in term of how well the components meet operationalizing goals represented in Figure 6. We firstly set the priority of each goal and then the degree that the component satisfies the goal. To illustrate, for our case study we set three different degree of satisfying: 0 - not supported, 1 - partly supported and 3 - completely supported. We then calculate the score for each vendor and we select the one that has the highest score (score = $\sum_{i=1}^{n} p_i.s_i$, p_i - priority of the goal i and s_i - degree of satisfying for the goal i). Figure 7 illustrates the selection process, and, in our case, it is Track Transport component from vendor1 will be selected.

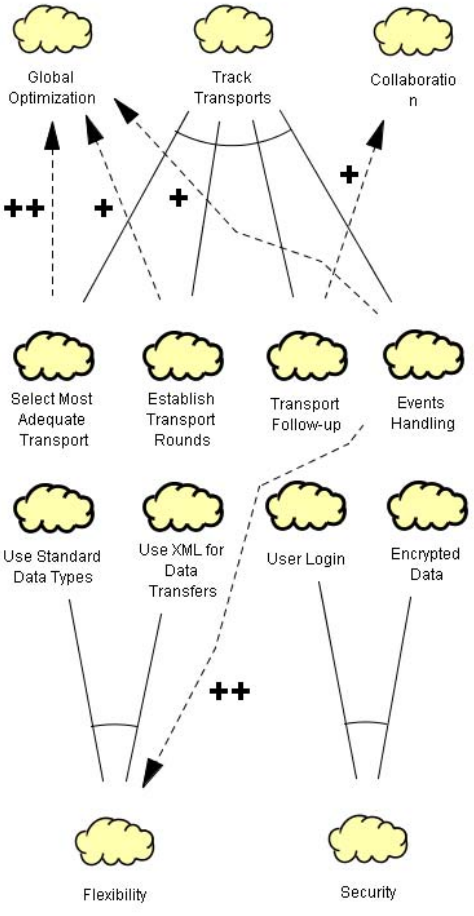


Fig. 6. NFR Goal-Graph for Track Transports

	Priority	Solution1	Solution2	Solution3
Select Most Adequate Transport	4	3	3	1
Establish Transport Rounds	3	3	3	1
Transport Follow-up	3	3	1	0
Events Handling	4	1	1	0
Use Standard Data Types	1	1	3	1
Use XML for Data Transfers	1	3	3	3
User Login	2	3	3	3
Encrypted Data	2	3	0	0
		50	**40**	**17**

Fig. 7. Evaluation of *Track Transport* component from three different vendors

5 Related Work

A number of COTS component-based software development methods have been proposed in literature. Some methods, such as OTSO (Off-The-Shelf Option) [10], STACE

(Social-Technical Approach to COTS Evaluation) Framework [11], and PORE (Procurement-Oriented Requirements Engineering) Method [12] emphasize the importance of requirements analysis in order to conduct a successful selection that satisfies the customer. But they do not support the complex process of requirements analysis and balancing with COTS features limitations. Our FaMOS-C approach emphasizes the importance of the functional and non-functional requirements of the system under development for identifying and selecting the COTS components. In addition, FaMOS-C provides a practical model for specifying the system requirements, matching components and ranking them. Our approach adopts and extends the concepts of some methods existing in the literature. Among them:

- [13] proposes the CRE (COTS based on Requirements Engineering) model for the COTS components selection. The CRE model focuses on non-functional requirements to assist the processes of evaluation and selection of COTS products. The CRE model adopts the NFR framework proposed in [14] for acquiring the user non-functional requirements;
- [15] proposes the REACT (REquirement-ArChiTecture) approach for the COTS components selection. The innovation, implemented in REACT, was to apply i* SD models to model software architectures in terms of actors dependencies to achieve goals, satisfy soft goals, consume resources and undertake tasks. The i* SD models are derived from existing use cases to represent the architecture of the system. The components are then selected to be plugged into the system architecture as instances of models actors.

The following major aspects characterize the novelty of our approach with respect to the existing literature:

- In FaMOS-C, COTS-Components are envisaged as services in the same way as custom-developed services. The two types of services are envisaged in the same models and analyzed specifically;
- FaMOS-C offers three views of the same reality using different levels of aggregation with custom purpose. The SSD offers an overview of both the organizational and component services of the application package that should be developed. The i* diagram describes the functional parts of each service. The NFR goal graph represents the functional and non-functional requirements of each service;
- For COTS component services, our FaMOS-C approach is to rank and select the components from different vendors in terms of how well the components meet the functional and non-functional requirements represented in the NFR goal graph.

6 Conclusions

Service-oriented architecture (SOA) promises to change the way business and technology vendors buy, sell, deploy, and manage application portfolios. For the first time, business users will be able to summon applications to support a business process rather than launch a business process constrained by the application. That is why the application of such technology into large hybrid software projects combining self-developed

services with third party vendors components is particularly interesting. In this context, we applied a service-oriented framework including a process for selecting a vendors' component to the development of a Supply Chain Management platform. Indeed, a supply chain actor requiring functionality to support any supply chain process that can be accessed directly from the vendor's library in the form of a service represents could be the business model of the coming years.

The developed framework offers three views of the same reality using different levels of aggregation with custom purpose. The strategic services diagram offers an overview of both the organizational and component services of the application package that should be developed. The i* strategic rationale diagram splits a service in a series of tasks, goals and resources so that the functional parts of the service can be enlighten. Finally, the NFR goal graph is used to find out the functional and non-functional aspect the service has to offer.

The framework remains however limited since, in its actual development, it only allows component selection. Work for enlarging its scope including component customization onto specific cases including adequate project management is in progress and will be the subject of a PhD thesis.

Acknowledgements. Most of the research on outbound logistics made at UCL/CESCM and the contents of this paper have been initiated by the Walloon region under the auspices of the TransLogisTIC project (www.translogistic.be). We gratefully acknowledge the region and the project industrial partners for their support.

References

1. TransLogisTIC: The translogistic project. Walloon Region (2006), http://www.translogistic.be
2. Wautelet, Y., Achbany, Y., Kolp, M.: A service-oriented framework for mas modeling. In: Proceedings of the 10th International Conference on Entreprise Information Systems (ICEIS), Barcelona (2008)
3. Pour, G.: Component-based software development approach: New opportunities and challenge. In: Proceedings Technology of Object-Oriented Languages, TOOLS, vol. 26, pp. 375–383 (1998)
4. Brown, A.W., Wallnau, K.C.: The current state of component-based software engineering. IEEE software, 37–46 (1998)
5. Pour, G.: Enterprise javabeans, javabeans & xml expanding the possibilities for web-based enterprise application development. In: Proceedings of Technology of Object-Oriented Languages and Systems, TOOLS, vol. 31, pp. 282–291 (1999)
6. Achbany, Y., Wautelet, Y., Kolp., M.: Process for developing adaptable and open service systems. Technical Report (2008)
7. Achbany, Y.: A multi-agent framework for open and dynamic service-oriented systems:investigation and application to web services (PhD thesis, Université catholique de Louvain, Louvain School of Management (LSM), Louvain-La-Neuve, Belgium)
8. Pache, G., Spalanzani, A.: La gestion des chaînes logistiques multi-acteurs: perspectives stratégiques. Presses Universitaires de Grenoble (PUG) (2007)
9. Samii, A.K.: Stratégie logistique, supply chain management: Fondements - méthodes - applications. Dunod (2004)

10. Kontio, J.: A cots selection method and experiences of its use. In: Proceedings of the 20th annual software engineering workshop, Maryland (1995)
11. Kunda, D.: e Brooks, L.: Applying social-technical approach for cots selection. In: Proceeding of the 4th UKAIS conference, University of York (1999)
12. Ncube, C., Maiden, N.A.M.: Pore: Procurement-oriented requirements engineering method for the component-based systems engineering development paradigm. In: International workshop on component-based software engineering (1999)
13. Alves, C., Castro, J., Alencar, F.: Requirements engineering for cots selection. In: The third workshop on Requirements Engineering, Rio de Janeiro, Brazil (2000)
14. Chung, L., Nixon, B., Yu, E., Mylopoulos, J.: Non-functional requirements in software engineering. Kluwer Academic Publishing, Dordrecht (2000)
15. Sai, V., Franch, X., Maiden, N.: Driving component selection through actor-oriented models and use cases. In: Kazman, R., Port, D. (eds.) ICCBSS 2004. LNCS, vol. 2959, pp. 63–73. Springer, Heidelberg (2004)

A Process for Developing Adaptable and Open Service Systems: Application in Supply Chain Management

Yves Wautelet, Youssef Achbany, Jean-Charles Lange, and Manuel Kolp

Louvain School of Management, University of Louvain, Belgium
{yves.wautelet,youssef.achbany,jean-charles.lange}@ucLouvain.be,
manuel.kolp@ucLouvain.be

Abstract. Service-oriented computing is becoming increasingly popular. It allows designing flexible and adaptable software systems that can be easily adopted on demand by software customers. Those benefits are from primary importance in the context of supply chain management; that is why this paper proposes to apply ProDAOSS, a process for developing adaptable and open service systems to an industrial case study in outbound logistics. ProDAOSS is conceived as a plug-in for I-Tropos - a broader development methodology - so that it covers the whole software development life cycle. At analysis level, flexible business processes were generically modelled with different complementary views. First of all, an aggregate services view of the whole applicative package is offered; then services are split using an agent ontology - through the i* framework - to represent it as an organization of agents. A dynamic view completes the documentation by offering the service realization paths. At design stage, the service center architecture proposes a reference architectural pattern for services realization in an adaptable and open manner.

Keywords: Requirements Engineering, Service-Oriented Modeling, i*, Multi-AgentSystems.

1 Introduction

Today's enterprise information systems have to match with their operational and organizational environment. Unfortunately, software project management methodologies are traditionally inspired by programming concepts rather than by organizational and enterprise ones. In order to reduce as much this distance, Agent-Orientation is increasingly emerging and has been the object of more and more research over the last 10 years. Its success comes from the fact that it better meets the increasing complexity and flexibility required to develop software applications built in open-networked environments and deeply embedded into human activities.

The gap between the traditional software engineering approaches and multi-agent software systems using artificial intelligent concepts remains nevertheless important. A series of development methodologies for designing multi-agent systems (MAS) have appeared over the years; those methods have their own characteristics and use various models to analyse the environment, to design the system and also various languages to implement the proposed solution. This paper is part of the effort to dispose of

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 564–576, 2009.
© Springer-Verlag Berlin Heidelberg 2009

methodologies covering the whole development life cycle: from the analysis of industrial (multi-actor) cases to the development of MAS software applications by extending the I-Tropos software process [1] notably through service orientation. This plug-in is the *Process for Developing Adaptable and Open Service Systems* (ProDAOSS, see [2,3]).

A supply chain is the set of all actors and relations between them which participate in the process of delivering value to a customer as a product or service. It includes all the processes from raw materials to delivery and is viewed as a network of information, materials and financial flows. In this paper we particularly focus on the application of our software development methodology on outbound logistics - i.e. the process related to the movement and storage of products and goods from the supplier to the end user - with a strong focus on transportation. The idea underlying the developed collaborative application package is to offer the most advanced services to optimize the transportation chain on a global basis. The paper will present the analysis and design stages of the application package development.

The paper is structured as follows. Section 2 discusses the research approach and main contributions of the paper. Section 3 briefly presents outbound logistics as envisaged in the study. Section 4 overviews the application development: the analysis and design stages are depicted in detail. Section 5 presents the related work. Finally, section 6 concludes the paper.

2 Research Approach and Contributions

This section describes our research approach and the contributions of the paper. Section 2.1 introduces the ProDAOSS process. Section 2.2 describes the notion of actor collaboration in supply chain management. Finally, Section 2.3 justifies the interest of a services approach in supply chain management.

2.1 ProDAOSS: A Methodology for Developing Service-Oriented MAS

Service-oriented computing is becoming increasingly popular when developing large and flexible software solutions. Since our goal is to dispose of a methodology for developing large industrial software application packages this was adopted in our process called ProDAOSS (i.e., Process for Developing Adaptable and Open Service Systems). Indeed, services are used as fundamentals to drive the software process. They are distinguished and documented at analysis stages (through the FaMOS framework, see [4]) then converted into service centers at design stage for an agent oriented implementation. More precisely:

- The organisation is firstly described as a set of organisational services in the Strategic Services Diagram;
- Organizational services are split in a series of goals, tasks and ressources depicted in the Strategic Dependency and Rationale Diagrams;
- Organisational services realization paths are documented by the Dynamic Service Hypergraph;
- Organisational services are designed as service centers in the architectural design discipline;

– Services' realization environment is open and adaptable through the use of a reinforcement learning alogrithm [5] and a probabilistic reputation model [6].

The contributions of this paper are the extension of the *Service Center Architecture* (first presented in [2,7,8], and extended here) and the instantiation of the whole Pro-DAOSS process to a case study in outbound logistics [1].

2.2 Actor Collaboration in Supply Chain Management

By nature, supply chain management is an interesting area for the development of industrial multi-actor software systems since it involves a series of collaborating or competing compagnies with tens of roles played by hundreeds of individuals. The benefits that can be taken from such systems can lead to avoid waste of resources for all the involved actors. Collaborative decision will tend to avoid local equilibriums (at actor level) and wastes in the global supply chain optimisation, giving opportunities to achieve the greatest value that the chain can deliver at lowest cost (see [9,10]). Such a result can only be achieved through actors' collaboration. As noticed by [11], the term collaboration is confusing since various interpretations take place in the context of supply chains, they distinguish 3 overlapping levels of collaboration in real supply chains:

– *Information Centralization* is the most basic technique of information sharing. Applied to outbound logistics it can be the shipper announcing its transportation needs or the carrier sharing all its planned transportations. Moyeu et al. distinguish information sharing from centralisation by the fact that the later is "*the multi-casting in real-time and instantaneously of the market consumption information*" [11] while the former is the sharing of the demand and supply information between companies;
– *Vendor Managed Inventory* (VMI) and *Continuous Replenshiment Program* (CRP) are collaboration techniques in which retailers do not place orders because wholesalers use information centralization to decide when to replenish them;
– *Collaborative Planning, Forecasting and Replenishment* (CPFR) enhances VMI and CRP by incorporating joint forecasting. CPFR includes only two levels of supply chain, retailers and wholesalers, it allows companies to electronically exchange a series of written comments and data which include past sales trends, scheduled promotions and forecasts. It shares more information than only demand information. This allows the participants to coordinate joint forecasts by focussing on the differences in forecasts.

Our interpretation of the term collaboration in the context of this paper is at first information centralisation, indeed by announcing the demand and supply concerning transportation (in terms of transport services, logistic requests, transportations, etc.), the actors collaborate for answering to the requirements of the other actors at best conditions. Indeed, collaboration on the basis of information sharing is not incompatible with competition. Since an actor role can be played by different - probably competing

[1] Outbound logistics is part of supply chain (management), this will be overviewed later in the paper.

- companies, the demander can use the best possible offer or the supplier can make interesting offers to dispose of optimally filled transportations. At second by sharing this information benefits can be issued in terms of global optimization. The whole outbound logistic chain can be optimized in terms of load balancing, cross dockings, tours, time, etc. Moreover, real-time information about the development of the services allows internal follow up and (re)optimization.

2.3 The Services Approach in Supply Chain Management

In the context of the development of a multi-actor collaborative application package we used a service-oriented approach for developing the MAS. This has been done following the ProDAOSS methodology (see [3,2] for a full description using the Software Process Engineering Meta Model (SPEM [12])). The services approach in the context supply chains is of particular interest because:

- Supply chain software and data collection will, in the future, take the form of a utility accessible on demand;
- Service-oriented architecture (SOA) could have a profound effect on the way supply chain software is designed, sold and implemented in the future. Common workflow based developments progressively leave place to generic customizable systems. In that perspective, SOA is designed to support rather than dictate business processes. This trend is for example also present in ERP systems development;
- Software developments' integration is from strategic importance in nowadays business context, SOA with its inherent flexibility ease these procedures.

2.4 MAS in Supply Chain Management: A Service-Center Approach

At first, a supply chain can naturally and easily be conceived as an organization of actors played by a series of companies. Each of them could be represented (i.e., instantiated) as one or many agent(s); agent-oriented modelling is thus particularly well indicated. Such modeling is achieved, in this paper, through an extension of i* [13]: the FaMOS framework [4].

Moreover, the Service Center Architecture (SCA) presented in this paper and applied in the context of outbound logistics envisages the dynamic allocation of tasks to competing or collaborating agents. Indeed, at design stage the application package is conceived as a multi-agent system through the use of the SCA which works as follow: when a service is requested by a particular agent (instance of the responsible actor documented in the Strategic Services Diagram), the series of task realizing the service (documented in the Dynamic Service Hypergraph) is communicated through the environment to the other agents so that task-specialist agents (documented in the Strategic Dependency and Strategic Rationale Diagrams) are assigned to the task(s) they are able to fulfil by a mediator agent evaluating the best possible candidate (following defined criteria such as reputation, cost, disposal of resources, etc.). This is documented in detail in Section 4.2.

Such an architecture is consequently of particular interest in outbound logistics, the aspect of supply chain management we choose to develop here and documented into the next section.

3 Outbound Logistics

Outbound logistics is the process related to the movement and storage of products from the supplier to the end user. In the context of this paper we mostly focus on transportation decisions, which will additionaly provide information for better internal storage decisions. The actors of the supply chain play different roles in the outbound logistic flow. The producer will be a logistic client in its relationship with the raw material supplier, which will be considered as the shipper. The carrier will receive transportation orders from the shipper and deliver goods to the client, while relying on the infrastructure holder and manager. In its relation with the intermediary wholesaler, the producer will then play the role of the shipper and the wholesaler will be the client.

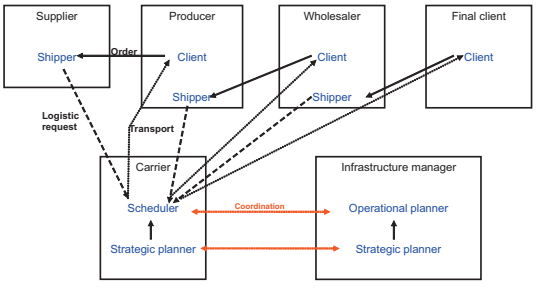


Fig. 1. Material flows in the outbound logistics chain

Figure 1 summarizes the material flows between the actors of the outbound logistics chain. The responsibilities corresponding to the different roles are:

- *Shipper*: has received an order from a client, and does a logistic request to a carrier for the delivery of that order.
- *Carrier*:
  - *The Strategic Planner*: decides on the services that are offered on the long term, on the use of infrastructure, on the logistic resources to hold and on the client's acceptation conditions.
  - *The Scheduler*: orders transports to be realized, according to the strategic network and constraints, coordinates with the infrastructure manager and assigns logistic requests to those transports such that the delivery requirements are met.
- *Infrastructure Manager*: holds the logistic infrastructure and coordinates with the carrier's scheduler to offer the network for the planned transports.

4 The Outbound Logistics Software Development: ProDAOSS Approach

This section introduces the application of the ProDAOSS process onto outbound logistics. The development of the outbound logistics application package is however too large to be fully developed here. That is why we will, in the Strategic Services Diagram, document the entire application package in terms of organizational services and focus on only one aspect, *Manage Transports*, in the rest of the presentation. So that, the Strategic Rationale Diagram, the Dynamic Service Hypergraph - at analysis stage - and the Service Center Architecture - at design stage - will only document this service.

4.1 Application Analysis

For the application analysis, the ProDAOSS process uses the models included in the FaMOS framework [4]. Firstly, we describe all the services included in the application package through the *Strategic Services Diagram*. Afterwards, the *Strategic Rationale Diagram* and the *Dynamic Service Hypergraph* focus on the service *Manage Transport*.

Strategic Services Diagram. The *Strategic Services Diagram* (SSD) of Figure 2 introduces all the services of the application package. This view allows all project stakeholders to share a common aggregate view of the services including their dependency relationships. As evoked previously, due to a lack of space, we will, in the context of this paper, focus on the *Manage Transports* service. To fulfil this service, the *Customer* depends of the *Scheduler*.

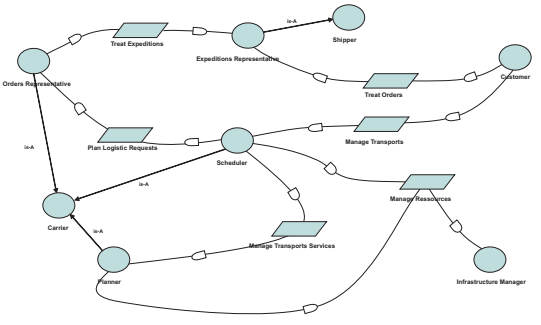


Fig. 2. Strategic Services Diagram for Outbound Logistics

Strategic Rationale Diagram. The *Strategic Rationale Diagram* (SRD) of Figure 3 depicts the actors, goals, tasks and resources involved into the *Manage Transport* service realization. This diagram offers a complete static view of the actors' dependencies for resources, goals and tasks. This view is from primary importance for designing the MAS since the agents that will instantiate the involved actors will be the ones that will be assigned the tasks realization into the Manage Service service center at design stage (more information is given into the next section). Seven actors are involved in the process of managing transports with the scheduler as the central one. The scheduler main task is to *plan a new horizon*: this is achieved by *creating transports* and assigning the *logistic requests* transmitted by the *Order Representative* to those transports (for more information see the realization paths in the paragraph below).

Dynamic Service Hypergraph. Finally, the *Dynamic Service Hypergraph* (DSH) offers a complementary view to the SRD presented above by showing the service realization paths. This diagram is also of primary importance in the context of the service center architecture at design stage (see next section).

Figure 4 represents fulfilment paths for the Manage Transport service. Each node is a step in service provision and each edge corresponds to the execution of a task t_k by a specialist agent $a^{SA}_{k,u}$, where u ranges over specialists that can execute t_k according to

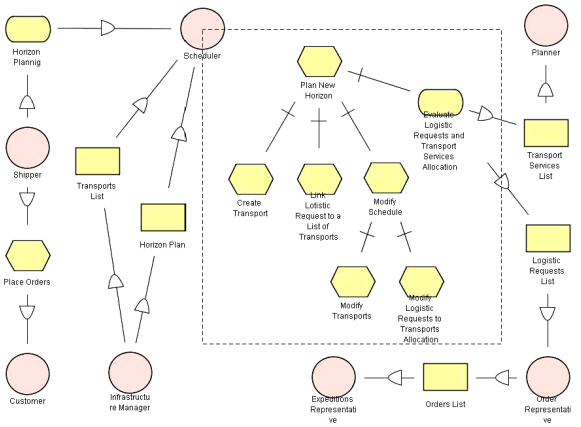


Fig. 3. Strategic Rationale Diagram for Manage Transport

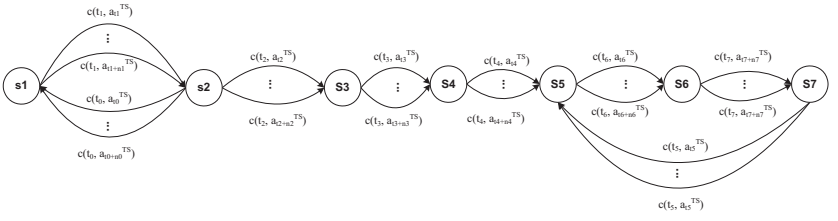


Fig. 4. Dynamic Service Hypergraph for Manage Transport

Table 1. Description of the tasks

Start	End	Task	Description
S_1	S_2	t_0	It modify requests allocation
S_1	S_2	t_1	It verifies the logistic requests or transportation services allocation.
S_2	S_3	t_2	It modifies requests allocation.
S_3	S_4	t_3	It creates transport.
S_4	S_5	t_4	It links the logistic requests to a series of transports.
S_5	S_6	t_5	It evaluates the schedule feasibility.
S_6	S_7	t_6	It modifies transports.
S_7	S_5	t_7	It modifies logistic requests attribution to transports.

the criteria set in the service request. A function of the criteria set $c(t_k, a_{k,u}^{SA})$, labels each edge and represents the QoS advertised by the specialist $a_{k,u}^{SA}$ for performing the task t_k. Note that different paths offering different QoS are available. Indeed, as shown in Table 1 path $< t_0, t_1, t_2, t_3, t_4, t_6, t_7 >$ offers alternative ways of fulfilling the service.

The following list gives a descritpion of each task used in the Hypergraph; tasks are here limited to a scheduler agent:

4.2 Application Design

The ProDAOSS process proposes, at architectural design stage, an open, distributed, self-organizing and service-oriented MAS architecture called *Service Center Architecture* (SCA) first defined in [2,8,7] and specialized/instancied here for an application to outbound logistics. SCA allows unbiased service provision driven by multiple concerns and tailored to user expectations. Deploying such a system places very specific criteria on the properties of MAS architectures that can support it:

1. A simple architecture would minimize the variety of interactions between the participating agents. Ensuring interoperability would thus not involve high cost on the providers.
2. Internal functions as task allocation, reputation computation, etc, ought to remain outside the responsibility of the entering and leaving agents to avoid bias. The architecture must therefore integrate a special class of agents that coordinate service provision and that are internal to (i.e., do not leave) the system.
3. Since there is no guarantee that agents will execute tasks at performance levels advertised by the providers, reputation calculation and task allocation should be grounded in empirically observed agent performance and such observation be executed by internal agents.
4. Varying quality of service (QoS) requests and change in the availability of agents require task allocation to be automated and driven by QoS, reputation scores and other relevant considerations (e.g., deadlines).
5. To ensure continuous optimization of system operation, task allocation within the architecture should involve continuous observation of agent performance, the use of available information to account for agent reputation or behavior, and exploration of new options to avoid excessive reliance on prior information.

A MAS architecture answearing those 5 criteria is proposed. This organizes agents into groups, called service centers. Each service center specializes in the provision of a single service. Within each center, a task allocation and reputation algorithm are integrated in the proposed architecture.

The suggested architecture, the task allocation algorithm and the reputation model allow the building of open, adaptable, distributed, service-oriented MAS, adaptable to changes in the environment by accounting on both experience acquired through system operation and optimal exploration of new task allocation options.

As presented in Figure 5, the logical architecture proposed in this section falls four layers.

Upper Layer: User Client. The upper layer represents user clients (humans or applications) that interact with the middle layer and transmit client requests.

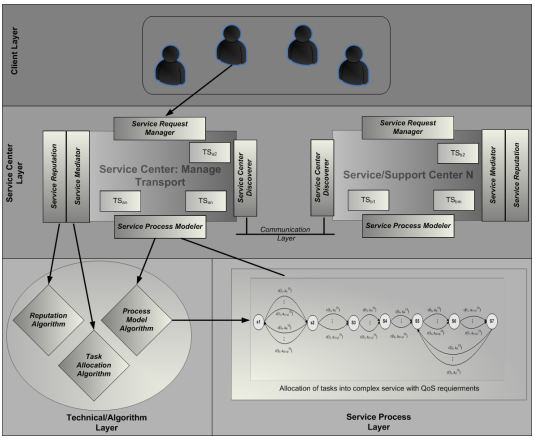


Fig. 5. Logical architecture and overview of the requierements-driven service-based architecture

Middle Layer: Service Center. The middle layer contains the various *service centers* based on the *Service Center Architecture* (see [2,7,8] for more details) connected by a communication layer. The Service Center Architecture (SCA) groups services into service centers (SC). Each SC contains all distinct *tasks*, executed by *Task-Specialist agents* (TS), needed to provide a *service* corresponding to a *service request* originating from the user. As represented in Figure 5, five special agents are also present in each SC:

- The *Service Request Manager* is responsible to receive and manage a service request. It represents the public interface of a SC. A service request gives: (i) a formula which describes the goal state or a set of goal states (that is, the important thing to do is to achieve any state in which this formula is true), (ii) a set of hard constraints on quantitative QoS criteria and (iii) a QoS criterion to optimize. When a request is received by this agent, it dispatches the request to the service mediator which allocates the different task to the given TS by optimizing the QoS criterion and by respecting the set of hard constraints.
- The *Service Mediator* composes tasks by observing past performance of individual TS, then subsequently using (and updating) this information through the task Allocation Algorithm (TA)(see [14,15,7]). Combining the SCA and the TA brings the following benefits: (a) Adaptability to changes in the availability and/or performence levels of TS is ensured, as the algorithm accounts for actual performance observed in the past and explores new compositions as new TS appear. (b) Continuous optimization over various criteria in the algorithm allows different criteria to guide service composition, while exploitation *and* exploration ensure Merdiator continually revise composition choices. (c) By localizing composition decisions at each Mediator, the architecture remains decentralized and permits distribution of resources. (d) The architecture and algorithm place no restrictions on the openness of or resource distribution in the system.
- The *Service Process Modeler* manages the hypergraph corresponding to the Dynamic Service Hypergraph that represents the execution of the service. Indeed, a

service can be understood as a process, composed of a set of tasks ordered over the graph representing the service. Each SC has a *Task Directory* (TD) which is a repository referencing the tasks available in this SC. Each entry of TD is a *Task Directory Entry* describing a task and all information about it.
- The *Service Reputation* is responsible to model and compute the reputation scores of all TS. To compute these reputations, it uses a reputation algorithms presented in [2,6]. The computation of the reputation scores is done on the basis of the feedbacks given by the users and these scores represent the internal quality of the TS in executing the corresponding task.
- The *Service Center Discovery* has the role to explore the network in order to discover the other SC to compose services from different centers in more complex services.

The SC representing the service *Manage Transport* and the corresponding Dynamic Service Hypergraph illustrating its process model are shown in Figure 5. For each task (i.e., each edge composing the Dynamic Service Hypergraph) of the service, we have a set of TS able to execute this task. For the *Manage Transport* service, we have seven tasks and thus seven set of TS. Each TS is characterized by a reputation score computed by the service reputation. The Mediator uses this reputation score and others quality constraints given by the users in their service requests in order to allocate tasks to fulfil the service.

Bottom Layer: Technical/Algorithm and Service Process. The bottom layers are the technical/algorithm layer and the service process layer. Technical/algorithm layer contains all the algorithms and technical processes used and executed by the special agent of the service center, i.e., the service process modeler executes a process model algorithm to build the service process corresponding to the global goal of its service center, the service mediator uses a task allocation algorithm [14,15,7] to allocate tasks to TS and the service reputation uses the reputation algorithm [2,6] to compute the reputation score of the TS. By keeping these algorithms and others technical processes in a lower level, we keep independent these technical features from the service center and these technical aspects can be easily changed and adapted without any changing to the higher level, i.e., the service center itself.

The service process layer contains business processes, i.e., compositions of tasks in simple services or compositions of simple services in a more complex service. For the service *Manage Transport*, this layer contains its process model represented by the Dynamic Service Hypergraph as shown in Figure 5. These layers communicate between each other in order to answer clients'needs.

5 Related Work

The application of MAS technology onto supply chain management has already driven plenty of literature. Indeed, the subject covers several aspects that is why we will focus here on the application of MAS development processes onto supply chain cases.

Penserini [16,17] focuses on the socially-driven approach of the Tropos software development methodology in order to build up an agent-based information system

prototype in supply chain management. Their approach however remains limited since their supply chain model is (too) simplified and mostly focuses on the application of architectural styles and design patterns at architectural and detailed design levels rather than on the whole development life cycle.

Govindu and Chinnam [18] proposes a generic process-centered methodological framework called *Multi-Agent Supply Chain Framework* (MASCF), to simplify MAS development for supply chains applications. Their idea is to map elements from the *Supply Chain Operations Reference* (SCOR, see [19]) model to the Gaia development methodology [20] for multi-agent supply chain system development. Mapping the SCOR and the Gaia concepts is interesting for analysts but it supposes modelling supply chain processes using this formalism while it would be strain forward to dispose of a *fully integrated* generic and flexible methodology for MAS development in supply chain management. That is why we choose to extend a current methodology with elements and axioms adapted for such flexible industrial developments rather than to map existing supply chain modelling elements to software development elements.

In Ghenniwa [21], the idea that of an electronic marketplace (eMarketplace) as an architectural model to develop collaborative supply chain management and integration platform is defended. In their architecture, the eMarketplace exists as a collection of economically motivated software agents of service-oriented cooperative distributed systems. The authors discuss how coordination approaches as auctions and and multi-issues negotiation can be developed in the context of this eMarketplace. The objective here is to enable business entities to obtain efficient resource allocation while preserving long-term relationships. This idea is also present in our software system but little developed here since the paper mostly focus on the methodology and its application on industrial cases. Our software system presentation at the light of their ideas would nevertheless constitute an interesting perspective.

6 Conclusions

Service-oriented architecture (SOA) promises to change the way business and technology vendors buy, sell, deploy, and manage application portfolios. For the first time, business users will be able to summon applications to support a business process rather than launch a business process constrained by the application. That is why the application of such technology into Supply Chain Management is of primary interest. Indeed, since the environment tends to be more and more competitive and rapidly evolving, flexibility in information systems functionalities is a natural trend of the marked. That is why a supply chain actor requiring functionality to support any supply chain process that can be accessed directly from the vendor's library in the form of a service could represent the business model of the coming years.

More pragmatically, this paper has presented the application of the ProDAOSS process i.e. a process for developing adaptable and open service systems onto an outbound logistics case study. The framework is conceived as a plug-in for the I-Tropos methodology so that it covers the whole development life cycle. In the context of this paper, only the analysis and design stages were presented. At analysis level, flexible business processes were modelled with different complementary views. First of all, an aggregate

services view of the whole applicative package is offered; then services are split using an agent ontology - through the i* framework - to design a flexible multi-agent system. A dynamic view completes the documentation by offering the service realization paths. At design stage, the service-center architecture proposes a reference architectural pattern for services realization in an adaptable and open manner.

The application package development in industrial context has already been partially completed. It has been conceived to work with a *Geographic Information System* (GIS) developed by a third party for real-time positioning of transports and dynamic horizon (re)planning in function of collected data. The conception using business services has empirically also been perceived to be fruitful into these multi-partners software developments.

Acknowledgements. Most of the research on outbound logistics made at UCL/CESCM and the contents of this paper have been initiated by the Walloon region under the auspices of the TransLogisTIC project (www.translogistic.be). We gratefully acknowledge the region and the project industrial partners for their support.

References

1. Wautelet, Y.: A goal-driven project management framework for multi-agent software development: The case of i-tropos. PhD thesis, Université catholique de Louvain, Louvain School of Management (LSM), Louvain-La-Neuve, Belgium (August 2008)
2. Achbany, Y.: A multi-agent framework for open and dynamic service-oriented systems:investigation and application to web services (PhD thesis, Université catholique de Louvain, Louvain School of Management (LSM), Louvain-La-Neuve, Belgium)
3. Achbany, Y., Wautelet, Y., Kolp., M.: Process for developing adaptable and open service systems. Technical Report (2008)
4. Wautelet, Y., Achbany, Y., Kolp, M.: A service-oriented framework for mas modeling. In: Proceedings of the 10th International Conference on Entreprise Information Systems (ICEIS), Bacelona (2008)
5. Sutton, R.S., Barto, A.G.: Reinforcement learning: An introduction. MIT Press, Cambridge (1998)
6. Fouss, F., Achbany, Y., Saerens, M.: A probabilistic reputation model. Technical Report (2008)
7. Jureta, I.J., Faulkner, S., Achbany, Y., Saerens, M.: Dynamic task allocation within an open service-oriented mas architecture. In: Proceedings of the 2007 International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2007) (2007)
8. Jureta, I.J., Faulkner, S., Achbany, Y., Saerens, M.: Dynamic web service composition within a service-oriented architecture. In: Proceedings of the International Conference on Web Services (ICWS 2007) (2007)
9. Pache, G., Spalanzani, A.: La gestion des chaînes logistiques multi-acteurs: perspectives stratégiques. Presses Universitaires de Grenoble (PUG) (2007)
10. Samii, A.K.: Stratégie logistique, supply chain management: Fondements - méthodes - applications. Dunod (2004)
11. Moyaux, T., Chaib-draa, B., D'Amours, S.: Supply chain management and multiagent systems: An overview. In: Chaib-draa, B., Müller, J.P. (eds.) Multiagent-based supply chain management, pp. 1–27. Springer, Heidelberg (2006)
12. OMG: Software and systems process engineering meta-model specification. version 2.0 (2008)

13. Yu, E.: Modeling strategic relationships for process reengineering. PhD thesis, University of Toronto, Department of Computer Science, Canada (1995)
14. Achbany, Y., Fouss, F., Yen, L., Pirotte, A., Saerens, M.: Optimal tuning of continual online exploration in reinforcement learning. In: Kollias, S.D., Stafylopatis, A., Duch, W., Oja, E. (eds.) ICANN 2006. LNCS, vol. 4131, pp. 790–800. Springer, Heidelberg (2006)
15. Achbany, Y., Fouss, F., Yen, L., Pirotte, A., Saerens, M.: Tuning continual exploration in reinforcement learning: An optimality property of the boltzmann strategy. Neurocomputing 71, 2507–2520 (2008)
16. Penserini, L., Kolp, M., Spalazzi, L., Panti, M.: Socially-based design meets agent capabilities. In: Proceedings of the IEEE/WIC/ACM International Conference on Intelligent Agent Technology, IAT 2004. IEEE CS Press, Beijing (2004)
17. Penserini, L., Kolp, M., Spalazzi, L.: Social oriented engineering of intelligent software. International Journal of Web Intelligence and Agent Systems (WIAS) 5, 69–87 (2007)
18. Govindu, R., Chinnam, R.V.: Mascf: A generic process-centered methodological framework for analysis and design of multi-agent supply chain systems. Computers and Industrial Engineering 53, 584–609 (2007)
19. SCC: Supply-chain operations reference-model (scor) - version 7.0 [overview]. Supply Chain Council (2005)
20. Wooldridge, M., Jennings, N., Kinny, D.: The gaia methodology for agent-oriented analysis and design. Autonomous Agents and Multi-Agent Systems 3, 285–312 (2000)
21. Ghenniwa, H., Dang, J., Huhns, M., Shen, W.: Emarketplace model: An architecture for collaborative supply chain management and integration. In: Chaib-draa, B., Müller, J.P. (eds.) Multiagent -based Supply Chain Management, pp. 29–62. Springer, Heidelberg (2006) (accepted)

Business Process-Awareness in the Maintenance Activities

Lerina Aversano and Maria Tortorella

Department of Engineering, University of Sannio, via Traiano 1, Benevento, Italy
aversano@unisannio.it, tortorella@unisannio.it

Abstract. In this paper we focus on the usefulness of business process knowledge for clarifying change requirements concerning the supporting software systems. To this aim the correctness and completeness of the change requirements impact were evaluated with and without the business process knowledge. Results of this preliminary empirical study are encouraging and indicate that the business information effectively provides a significant help to software maintainers.

Keywords: Software system evolution, Impact analysis, Business process modelling, Empirical study.

1 Introduction

Fast change in business requirements force enterprises to a continue evolution of their software systems for effectively using them. Changes emerging from the business environment immediately affect the business processes needing to be customized to support organizational change [2, 5]. Moreover, technological updates and innovation also affect the way business is carried out. Software systems provide a support to the users while efficiently performing their activities. As a consequence, supporting the software maintenance tasks for adapting the software systems to the business process changes is an emerging challenge. A business process consists of the activities performed by an enterprise to achieve a goal. Its specification includes the activity description and their control and data flow (input received by each activity / output produced by each activity). The supporting software system is generally an application providing a support to the user while performing these activities. The support can be provided to all the process activities or just a subset of them. Then, it is clear that a change in the process may immediately affect the software components. However, locating the components impacted by the change requirements is often not obvious to a software maintainers. This is particularly true if the change requirement is expressed in terms of business activities and the maintainers is not provided with such an information. This immediately suggests the need of adequately managing the link between business processes and software systems. But, an adequate way to proceed is achieving an evidence of this need. With this in mind, an empirical study proposed in this paper was performed. In particular, the aim was to demonstrate the usefulness of knowledge of a business process for clarifying change requirements concerning the supporting software systems. To this aim, the *correctness* and

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 577–589, 2009.
© Springer-Verlag Berlin Heidelberg 2009

completeness of the change requirement impact were evaluated with two different approaches: one using just the software documents; and one using also the business process knowledge.

The research question investigated in the empirical study was: Does the knowledge regarding the business process help to better identify the impact of a requirement change request on the supporting software system?

Results of this preliminary empirical study are encouraging and, according to our hypothesis, indicate that the business information effectively provides a significant help to software maintainers.

The paper is organized as follows: Section 2 presents the business process knowledge used in the maintenance tasks; Section 3 presents the empirical study design with the research questions and used case studies; Section 4 provides a description of the results supported by a statistical validation and analysis of the most interesting graphs; Section 5 gives an overview of related works; the subsequent section discusses threats of validity; concluding remarks and future works are discussed in Section 7.

2 Business Knowledge in Software Maintenance

The connection between a business process and supporting software systems is a relevant knowledge for software maintainers while they have to deal with change requirements. This connection is very often not adequately documented and the impact of a change in the process is difficult to map to the software components. In the best of our knowledge, there is a lack of empirical studies aiming at giving evidence of the relevance of this connection and, as a consequence, the need of methods and tool to manage it.

It is worth noting that, enterprises have to address continuously changes in their business processes and, as a consequence, on the supporting software systems. Nevertheless, software engineers do not document the design of a system with an explicit reference to the business process where they have to be used.

In the empirical study presented in this paper, software maintainers are provided with an explicit documentation at design level of the relationships existing between the business process and the involved software system. In particular, the documentation additionally produced consists of:

- a description of the business process in terms of activity diagram; and
- a corresponding matrix for documenting the links between process activities and software components.

Business process definitions can be specified using various languages, such as BPMN (Business Process Modeling Notation Specification) [12] and XPDL (XML Process Definition Language) [13]. However, this is a textual description of the process, which is necessary when there is a need to perform an automatic mining of it. Our purpose is different, as we are not interested to automatically manage the process definition, instead our aim is to make the software maintainers aware of the model of the business process adopting the software system they are working on. Therefore, we decided to use UML Activity Diagrams for representing the business processes.

Figure 1 shows the activity diagram documenting one of the business process supported by one of the systems used in the considered case studies. The activity diagram is provided with a textual description of the process. Essentially, the following information is specified:

Activity Description: to describe at a lowest level of details the human tasks performed.
Data Flow: to describe the inputs and outputs data required to perform the activity.
Control Flow: to define the flow among the process activities, including sequences, alternatives, iterations, parallel and pre/post conditions.

Figure 1 describes the business process used by a voluntary association, named *Beneslan*, to manage object donations for needy children. The software system used by the association to manage the donated objects and their distribution is named SantaClaus[1].

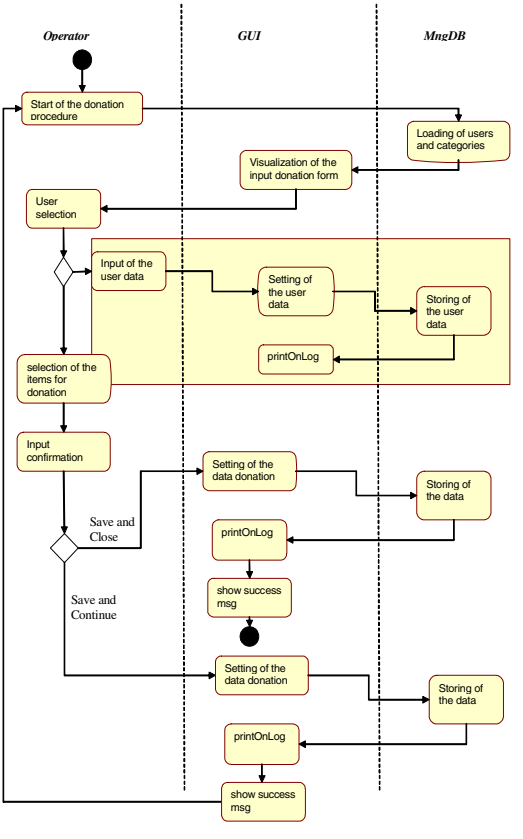


Fig. 1. Model of the business process where SantaClaus is used

[1] http://santaclaus.beneslan.it/santaclaus/

Table 1. Fragment of the traceability matrix between business process activities and system components

		post_article activity																notification activity															
		Start of the donation procedure	Loading of users and categories	Visualization of the input donation form	User selection	Input of the user data	Setting of the user data	Storing of the user data	printOnLog	Input of the donation data	Input confirmation	Setting of the data donation	Storing of the data	printOnLog	show success msg		Request reception	loading of available items	visualization of the available items	selection of the items for donation	donation confirm	visualization of the receiving user data	filling form receiving user	end donation	setting item data	storing item modification	New receiving user creation	New user storing	printOnLog	
ss	method																																
cle	index																	x	x	x	x												
	indexUser																x																
	search																	x	x														
	searchUser																	x	x														
	post_articoloUser		x							x																							
	save_posted										x		x																				
	save_Edit																									x							
	edit																	x	x	x					x								
	remove									x																							
	remove_user																																
	assignation																				x	x	x										
	saveAssignation																																
										x																							
vince	getProvincesList					x																											
nune	getListByProvince					x																											
	returnListByProvince																																

The second part of the documentation is the traceability matrix reporting the connection between activities in the model of Figure 1 and software components of SantaClaus. A fragment of this kind of table is shown in Table 1. The table shows an explicit reference between the system components used in each activity of the process. In particular, the link with the software system has been documented at two levels of granularity considering the class and methods.

3 Design of the Study

The empirical study aimed at obtaining two kinds of information: a quantitative one regarding how helpful is the knowledge regarding the business process supported by a software system when this one has to be evolved; and a qualitative one aimed at establishing how adequate the empirical environment and clear the provided documentation were. The aim was to understand if the inadequacy of the experimental conditions could influence the empirical results. In the following, a description of the two parts follows.

The subjects involved in the empirical study were students from the courses of Enterprise Information Systems and Software Project management, in their last year of the master degree in computer science at the University of Sannio in Italy.

The empirical study aimed at evaluating whether the use of process knowledge leads to an improved comprehension of the impact of a requirement change request on the software system. If this hypothesis proves valid, it will demonstrate that considering the business context where a software system is used is of crucial importance for effectively maintaining it.

The *treatment variable* of the empirical study was the independent variable *approach adopted*, the values being: *Business Knowledge approach*, and *No Business Knowledge approach*. The first value regarded the approach exploiting the knowledge

concerning the business process supported by the software system to be evolved, for analysing it and identifying the components impacted by the requirement software changes. The second value regarded the approach not using such a kind of knowledge.

The two approaches were applied by two different groups of 6 subjects each, randomly grouped. The first group, indicated as *G_BP*, was asked to use the *Business Knowledge approach*, while the second group, *G_NBP*, had to apply the *No Business Knowledge approach*. For each session, the subjects of both G_BP and G_NBP was supplied of the documents of Software Requirement Specification, Design Description, and code, and requirement change requests. Just G_BP was also furnished with the models of the business processes using the software systems to be evolved and traceability tables connecting business activities and software components.

The *software system to be analysed* was the second independent variable considered. Two software systems were analysed in two different sessions. Santaclaus is the web application, described in the previous section. It was written in PHP and Java. Uniflight is a software system, written in PHP and C, aiming at supporting the activities of an airlines for both ticket booking and managing trips, personnel, luggage and so on.

Table 2. Change requests

ID	Change Request	# impacted software components
Software system: **Santaclaus**		
CR1	Eliminate the category of the donated object and leave just a description	6
CR2	Allow other associations to register and use the site	2
CR3	Include the opportunity to communicate the donor when the objects he donated were assigned to somebody needing it	1
CR4	Include the functionality of automatically matching required objects with those available	2
Software system: **UniFlight**		
CR1	Include a new functionality to notify urgent communications and offers to the registered users	2
CR2	Include the possibility to manage companies as user of the system	2
CR3	Construct a list of flight with price lower than a given value	6
CR4	Offer the opportunity to choose different kinds of payment for the bought tickets, such as bank transfer	2

During the sessions, the subjects could use the time they wished for analysing the assigned software systems and accomplishing the assigned tasks. Two software systems were used for understanding if, after adequate training, the evaluation of different software systems gave the same results, regardless of their complexity and application domain.

Another independent variable was the *requirement change request*. Four requirement changes were requested for each system. They were labelled *CR1*, *CR2*, *CR3* and *CR4*, and each empirical subject was asked to face them in the established order. This variable was useful for understanding if the knowledge regarding the software system and gained by analysing a change request could influence the analysis of the next ones. Table 2 lists the change requests for the two software systems. Two dependent variables were calculated: *Correctness* and *Completeness.*

Correctness indicates how *correctly* each empirical subject identified the impacted software components. It is calculated as the proportion of the impacted software components correctly identified, *#CorrectSWComp*, on the total number of identified components, *#IdentifiedSWComp*. Table 2 lists the correct number of software components really impacted by each change request. The correctness is evaluated as indicated in the following formula:

$$Correctness = \frac{\#CorrectSWComp}{\#IdentifiedSWComp}$$

Completeness measures how *completely* the impacted software components were identified. It is evaluated as the proportion of the impacted software components correctly detected by each subject, *#CorrectSWComp*, on the total number of the impacted components, *#TotalSWComp*. It is computed as:

$$Completeness = \frac{\#CorrectSWComp}{\#TotalSWComp}$$

The *correctness* and *completeness* indicate the *usefulness* of the two approaches. *Completeness* also expresses the level of exploitation of the available documentation; the more complete the identification of the impacted components, the better the available information are *exploited* during the analysis.

The values of the two dependent variables obtained within G_BP and G_NBP were compared to identify the most effective method for identifying the software components impacted by a change request. Therefore, the hypothesis considered are the following:

H_0: there is no difference between the effectiveness of the approach using the business knowledge and the one not using the business knowledge.

H_a: there is a difference between the effectiveness of the approach using the business knowledge and the one not using the business knowledge.

The expected results were that H_0 could be rejected, i.e. the approach using the business knowledge is the most effective.

Table 3. Agreement questionnaire

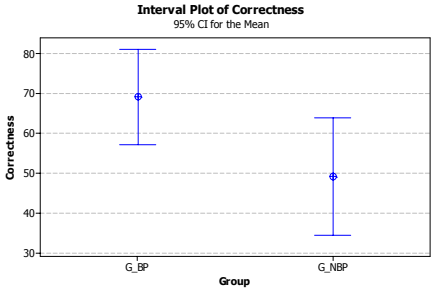
Common questions	
ID	Questions
Q1	How long did you take for performing the task?
Q2	Did you have enough time to perform the session task?
Q3	Was the session goal clear?
Q4	Were the change requests clear and well stated?
Q5	Did you have difficulties to understand the documentation?
Extra questions	
ID	Questions
Q6	How long did you spend for understanding activity diagram and related software components matrix?
Q7	How long did you spend for understanding the rest of the documentation?
Q8	Were the activity diagrams clear?
Q9	Were the activity diagrams useful?
Q10	Was the business-software traceability matrix clear?
Q11	Was the business-software traceability matrix useful?

Table 3 lists the questions asked in the end of each session. The first five questions were asked to both groups and aimed at understanding how the subjects considered the provided documentation. Questions from Q6 to Q11 were posed just to G_BP and aimed at understanding the usefulness and exploitation of business knowledge. Questions Q1, Q6 and Q7 could be answered on the basis of the scale: <30min, 30-60 min, 60-90 min, > 90. Question Q2-Q5 and Q8-Q11 were responded by using a scale going from 1 to 5 on the basis of the level agreement.

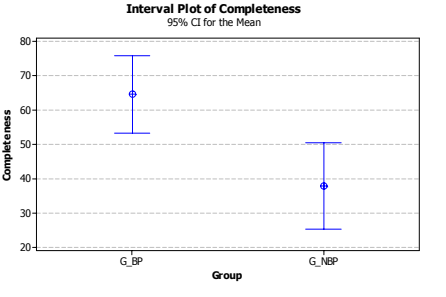
4 Results

Figure 2 shows that G_BP achieved better values of both correctness and completeness than G_NBP. In particular, Figure 2b shows that the worst value of completeness of G_BP is higher than the best value reached by G_NBP. This should indicate that applying the technique using the knowledge regarding the business process helps to obtain better results.

Figure 3 confirms the full results and details them for the singular case study. The figure highlights that both correctness and completeness have a better value for the G_BP in both cases. Still, the same results can be obtained when the singular change requests are observed. Figure 4 highlights these outcomes. G_NBP achieved a better

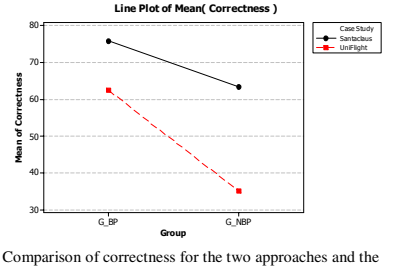


(a) Comparison of correctness for the two approaches

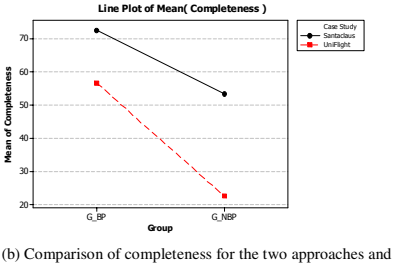


(b) Comparison of completeness for the two approaches

Fig. 2. Results of the approach application



(a) Comparison of correctness for the two approaches and the case studies



(b) Comparison of completeness for the two approaches and case studies

Fig. 3. Results of the approach application in the two case studies

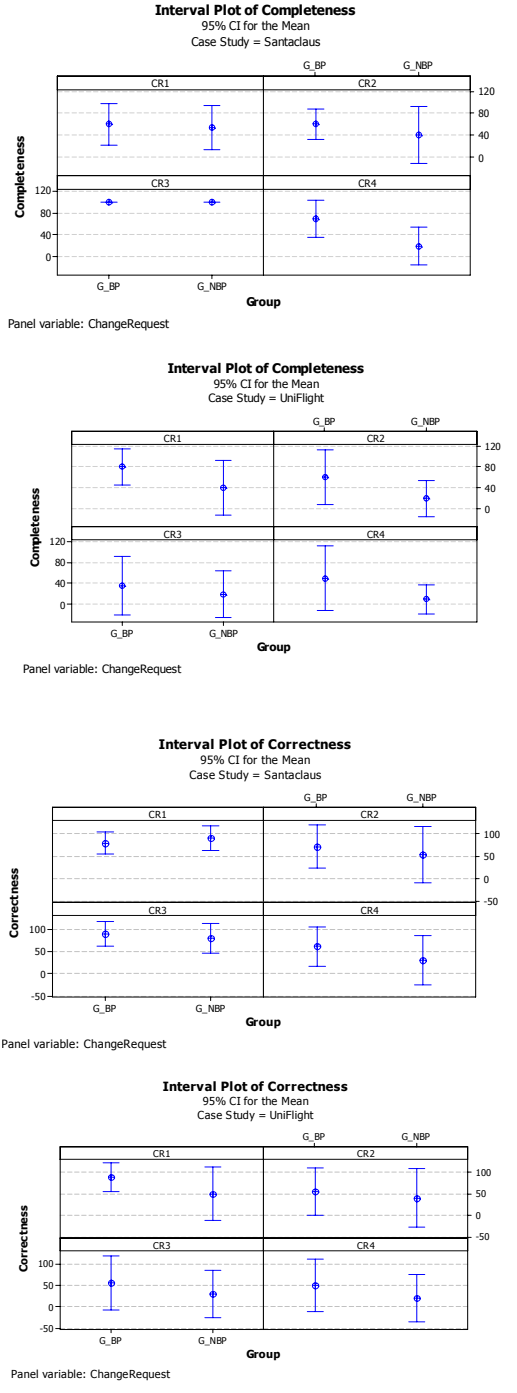


Fig. 4. Correctness and completeness results for each change request

results only for the first research question, CR1, in the first case study, Santaclaus. This is justified as the subjects using the business approach at the first application were concentrated on how the additional business knowledge had to be used. This gap was overcome with the next uses of this knowledge and Figure 4 shows that in all cases the Business approach gives better results for both correctness and completeness.

For understanding if the importance of the effect of the approach using the business knowledge, the one-way ANOVA analysis and, where needed, two-way ANOVA analysis, were executed [6]. Tables 4, and 5 show the most meaningful results.

The values reported in the ANOVA tables are: the degree of freedom effect (*DF*), the sum of squares (*SS*), the mean squares effect (*MS*), the mean squares error (*MS error*), the value of the F statistic *(F)* and the associated probability (*p-value*) to test the null hypothesis. A *p-value* of 0.05 is the most commonly accepted threshold. A *p-value* of less than 0.05 means that the null hypothesis, H_0, may be rejected and thus that the independent variable has a significant effect.

Tables 4 and 5 show that for the two dependent variables, the treatment variable has a low error probability if H_0 is rejected. Therefore, the approach using the business knowledge positively affects the effectiveness of the evaluation. The tables underline that no difference exists between the results obtained when the G_BP group analyses a software system (p-value > 0.05, when G_BP analyses both Santaclaus and UniFlight). This underlines that the results obtained by applying the business knowledge helps to reach some results independently from the application domain. On the contrary, a significant effect of the analysed software systems exists on the results of group G_NBP. To summarise these results indicate that using business knowledge helps to reach better outcomes than the other one, and their goodness does not depend on the application domain.

Table 4. Anova analysis for Correctness

Independent Variables	Couples of compared components	*DF*	*SS*	*MS*	*F*	*p-value*
Approach adopted	*Business vs NoBusiness*	1	7981	7981	4.53	0.036
Case study in GroupA	*Santaclaus vs Uniflight*	1	1803	1803	1.30	0.262
Case study in GroupB	*Santaclaus vs Uniflight*	1	8028	8028	4.08	0.050
Case Study – Approach adopted		1	1111	1110.80	0.66	0.418

Table 5. Anova one-way analysis for Completeness

Independent Variables	Couples of compared components	DF	SS	MS	F	*p-value*
Approach adopted	Business vs NoBusiness	1	14222	14222	10.08	0.002
Case study in GroupA	Santaclaus vs Uniflight	1	2507	2507	2.03	0.163
Case study in GroupB	Santaclaus vs Uniflight	1	9507	9507	7.08	0.011
Case Study – Approach adopted		1	1125	1125.0	0.87	0.353

In order to analyse the interaction between the treatment and the case study variables, it was necessary to perform the two-way analysis. Tables 4 and5 show, in the last row, that there is no significant interaction between the approach adopted and the case studies.

Table 6. Agreement questionnaire results

Value / Question	1	2	3	4	5
Q_2	19	5	0	0	0
Q_3	12	9	2	1	0
Q_4	7	14	2	1	0
Q_5	8	11	5	0	0
Q_8	6	4	2	0	0
Q_9	2	3	3	3	1
Q_{10}	6	1	5	0	0
Q_{11}	8	2	1	0	1

The agreement-questionnaire of the qualitative analysis helped to recognize that both groups, G_BP and G_NBP, spent the same time to conclude the assigned task. This allowed to conclude that the analysis of the business documentation does not require additional time, or, if it does, less time is spent for analysing the rest of the documentation.The agreement level to the other questions listed in Table 3 are shown in Table 6. The second bold line separates the questions common to both groups from those specific to G_BP. The data included in the table refers both case studies. The table shows that almost all empirical subjects agreed on the clarity of the aim of the study (Q3) and asked analysis requests (Q4). Almost all agreed on their confidence in the documentation (Q5), indicating its quality and comprehensibility. The answers to the specific questions show that, nevertheless the better results obtained in G_BP, aim and way to use of the activity diagrams and business-software traceability matrix were not clear for everybody. The reason is that no technique was suggested for reading and exploiting this kind of knowledge, but this was left to the experience and wisdom of the empirical subjects.

5 Related Work

Some research activities were focused on the relationships between the business processes and embedded software systems when the organization exploiting them is evolving. In [5], the authors underline the role of information technology in the move towards a process oriented view of the management. They argue that, besides other factors influencing the outcome of Business Process Re-engineering (BPR) projects, such as top management support and project management, legacy software systems have a critical impact. Moreover, they present a framework for understanding BPR, which is based on a study of 12 European organizations from a cross section of

industries dealing with BPR projects. From their study they evidenced that the outcomes of BPR strategies are influenced by the state of legacy software systems.

In [3], the authors suggest an integral view of business process re-engineering, based on both strategic and technological needs at the operational level. The software re-engineering activity aims at using existing corporate resources in a more economic way, i.e. reusing application knowledge to maximum extent, while software engineering approaches concentrate on handling software development according to the end-user requirements within a given amount of time and with limited cost.

The relationships among organizational and process aspects, and software systems have already been considered with reference to the development of new software systems [1, 4, 7, 10, 11]. These studies focus on capturing the organizational requirements for defining how the system fulfils the organization's goals, why it is necessary, what are the possible alternatives, etc. A developed technique, referred to as i* [10, 11], represents these aspects. i* offers two models to represent organizational requirements: the Strategic Dependency Model and the Rationale Dependency Model. The former focuses on the intentional relationships among organizational actors, while the latter permits one to model the reasons associated with each actor and their dependencies.

Besides i* [1, 4, 7, 10, 11], a family of goal-oriented requirements analysis (GORA) methods, such as KAOS [4] and GRL [1], have been proposed as top-down approaches for refining and decomposing the customers' needs into more concrete goals that should be achieved for satisfying the customers' needs. However, all these methods concern the definition of the requirements for the development of software systems.

More recently some research work are addressing the recovery of Business Processes and Design Documents.

Wil van der Aalst et al. [8, 9] use dynamic analysis to monitor the events generated from workflow management systems in order to recover business processes. In comparison, we use static analysis of the source code of the UI and business logic tiers of business applications which can be developed without using the workflow management systems. In the future, we plan to leverage tools developed by Wil van der Aalst *et al.* to verify the results of our static analysis.

6 Threats to Validity

The types of threats to the validity of the presented empirical study are: internal, external, and construct validity.

Internal Validity - The main issue related to internal validity, is due to the design of the empirical study and in particular to the possible information exchange among the empirical subjects between the sessions. This is avoided as the subjects were not allowed to communicate with each other during sessions, and were required to hand in all material at the end of each session.

External Validity – Similarly to any other academic empirical study another threats to validity is the issue of whether the subjects are representative of software professionals. Even if, in this case, the empirical subjects were student of the master degree in computer science and, than, were probably better trained at software

modeling with the UML than most software professionals. Another external validity issue, which is unfortunately inherent to controlled experiments, is the size and complexity of the system models.

Construct validity - Construct validity is related to the limitation of the performed measurements: in this case the completeness and correctness; additional parameters could be used for evaluating the impact of the maintenance task.

7 Conclusions and Future Work

Evolving software systems for satisfying requirements change needs requires the analysis and assessment of software documentation. A valid help for understanding software system functionality and requirement change requests can come from the analysis of the business knowledge coded in a used software system. Such kind of knowledge can be observed in the business processes adopting the software systems to be evolved. With this in mind, an empirical study has been conducted regarding the analysis of the impact of a set of requirement change requests on the components of two software systems. The study compared the results obtained by exploiting the knowledge of the business processes using the software systems with those ones obtained without using such kind of knowledge.

The obtained results confirmed the formulated hypothesis that the use of the business knowledge permits to reach better values of correctness and completeness in the identification of the impact that a change request may have on a software system. These outcomes were reached even if the empirical subjects did not know any technique for exploiting the business knowledge, but this was left to their experience and wisdom.

These results encourage to continue investigating in the role that the business knowledge has in the software engineering techniques. In particular, methods and techniques specialized for recovering and exploiting this kind of knowledge will be explored and defined.

In addition, the authors will continue to search for additional evidence of their hypothesis with additional studies also involving subjects working in operative realities.

References

1. Antón, A.I.: Goal-based requirements analysis. In: Proceedings IEEE International Conference on Requirements Engineering (ICRE 1996). IEEE Computer Society Press, Los Alamitos (1996)
2. Aversano, L., Bodhuin, T., Tortorella, M.: Assessment and Impact Analysis for Aligning Business Processes and Software Systems. In: Proceedings of ACM Symposium on Applied Computing (SAC 2005). ACM Press, New York (2005)
3. Bernd, J., Clifford, T.Y.T.: Business process reengineering and software systems strategy. Technical Report n.11, Institut für Wirtschaftsinformatik, Universität Tübingen, Tübingen DE (February 13, 2009), http://www.uni-tuebingen.de/wi/forschung/Arbeitsberichte3/ab_wi11.ok/ab_wi11.pdf

4. Dardenne, A., van Lamsweerde, A., Fickas, S.: Goal-directed requirements acquisition. Journal of Science of Computer Programming 20(1), 3–50 (1993)
5. Light, B., Holland, C.P.: The influence of legacy information systems on business process reengineering strategy. In: Proceedings of Business Information Management—Adaptive Futures (BIT 1998). Manchester Metropolitan University, Manchester (1988)
6. Lindman, H.R.: Analysis of variance in complex experimental designs. W. H. Freeman & Co., New York (1974)
7. Mylopoulos, J., Chung, L., Yu, E.: From object-oriented to goal-oriented requirements analysis. Communications of the ACM 42(1), 31–37 (1999)
8. van der Aalst, W.M.P., de Beer, H.T., van Dongen, B.F.: Process Mining and Verification of Properties: An Approach Based on Temporal Logic. In: Proceedings of International Conference on Cooperative Information Systems. Springer, Heidelberg (2005)
9. van der Aalst, W.M.P., Herbst, J.: Workflow Mining: A Survey of Issues and Approaches. Data and Knowledge Engineering 47(2), 237–267 (2003)
10. Yu, E.S.K.: Modeling organizations for information systems requirements engineering. In: Proceedings of International Symposium on Requirements Engineering (RE 1993). IEEE Computer Society Press, Los Alamitos (1993)
11. Yu, E.S.K.: Modelling strategic relationships for process reengineering. Doctoral Dissertation, Department of Computer Science. University of Toronto, Ontario (1995)
12. Business Process Modeling Notation Specification (February 13, 2009), http://www.bpmn.org/Documents/OMGFinalAdoptedBPMN1-0Spec06-02-01.pdf
13. Workflow Process Definition Interface – XML Process Definition Language (February 13, 2009), http://www.wfmc.org/standards/docs/TC-1025_10_xpdl_102502.pdf

BORM-points: Introduction and Results of Practical Testing

Zdenek Struska and Robert Pergl

Department of Information Engineering, Faculty of Economics and Management
Czech University of Life Sciences, Kamycka 129, Prague 6, Czech Republic
struska,pergl@pef.czu.cz
http://kii.pef.czu.cz

Abstract. This paper introduces the BORM-points method. The method is used for complexity estimation for information systems development. In the first part of the paper there is a detailed description of BORM-points and its specifics. In the second part there is a presentation of results of BORM-points application for real projects.

Keywords: Complexity estimation, Use case points, BORM method, Analysis and design of information systems, Complexity, Design phases of information system, BORM-points, Technical factor, Environment factor, Customer factor and Productivity factor.

1 Introduction

The area of complexity estimation is becoming more and more important because software systems of greater complexity are being designed and implemented. At the beginning of software development it is very important to know an estimation of how much the final software will cost. The knowledge of the software costs is required for deciding about proceeding with the development.

The first and the mostly used is the approach of an expert estimation. The approach is based on knowledge of an experienced IT expert or typically several experts based on their experience. Another approach is estimation based on model. The most well-known methods in this category are Function Points Analysis [1], Feature Points Analysis [12] and Use Case Points (UCP) [5].

The introduced BORM-points (BORMp) method is inspired by the UCP method. It eliminates some disadvantages of UCP and enhances the original method by the "customer factor".

BORMp estimates the development effort for information systems that are designed using the BORM method (Business and Object Relation Modelling) [2], [11]. BORM is not only applicable for software design, but it may be used for requirements analysis of a planned system and business process modelling as well. The method has been developed since 1993. BORM focuses on the pure object-oriented programming languages and systems as Smalltalk or object databases.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 590–599, 2009.
© Springer-Verlag Berlin Heidelberg 2009

2 The BORM-Points Method

The proposed BORMp method is based on the calculation of UCP [11]. UCP is a well-known method used for complexity estimation of information systems development. BORMp uses some parts of UCP, which are useful for the BORM methodology.

The new BORMp parts are designed to eliminate some known disadvantages of UCP, mostly a small customer orientation. It may, however, influence project complexity very much.

BORMp tries to estimate the complexity based on selected components characteristic for BORM. Calculation is divided into two independent steps. In the first step the number of participants and the number of business diagrams are counted. The second step consists of technical, environmental and customer factors evaluation.

2.1 Complexity Estimation Using BORMp

The proposed calculation of complexity estimation using BORMp is divided into two steps, because it is necessary to separate the *unadjusted number* that comes from the reality model and technical, environmental and customer factors that represent external influences on the system.

The unadjusted number is computed in the first step. In the second step individual technical, environmental and customer factors are computed:

- Unadjusted part of BORM points,
  - Number of participants,
  - Number of business diagrams.
- Technical factor.
- Environment factor.
- Customer factor.
- Productivity factor.

The structure of BORMp computations is in Fig. 1.

2.2 The Unadjusted Part of BORMp

The unadjusted part represents clear specification of number of participants and business diagrams. The theory of unadjusted part was published in [11]. For better comprehension, the summary follows.

Unadjusted Participant Weights – *uapw*. The participants are external objects that have some sort of relationship with the modelling system. They may be the system users, cooperating users, other software systems, etc. Participants should be described in the project documentation.

In BORMp the participants are divided by their measure of complexity:

simple–average–complex.

Simple – systems with automated interface to the measured system.

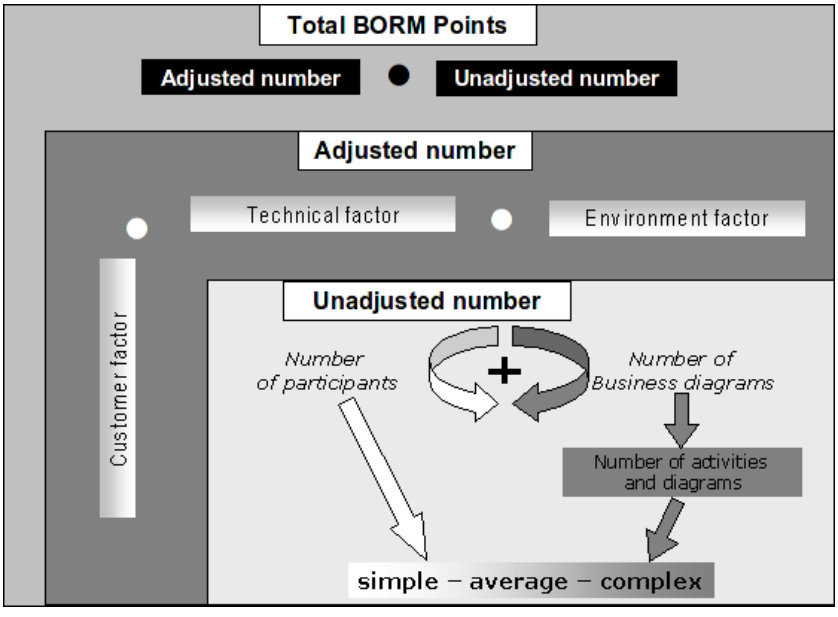


Fig. 1. Structure of BORM points calculation

Average – systems (such as data warehouse) connected to the measured system by a protocol (e.g. TCP/IP) or through a user interface.

Complex – participants cooperating with the system through a manual interface (mostly final users).

The participant categories and their corresponding weights are in Tab. 1.

Table 1. Weights of participants categories

Participant Type	Definition	Weight
Simple	Automated interface	1
Average	Interactive or protocol-driven interface	2
Complex	Manual interface (final users)	3

The weighted number of participants in each category is counted after the participants categorisation: simple participants get the weight 1; average get the weight 2 and the complex get 3. The summed result is the total unadjusted participants weight *upw* (see the Tab. 2).

Unadjusted Business Diagram Weights – *ubdw*. The BORM method is used for wide area of process mapping (including IT analysis). It is thus necessary to identify business diagrams related directly to the designed information system. Business diagrams get complexity weight according to their number of activities and/or transactions.

Table 2. Total unadjusted participants weight

Participant Type	Participant Weight	Number of Participants	Total
Simple	1	s	$1 \times s$
Average	2	a	$2 \times a$
Complex	3	p	$3 \times p$
Total unadjusted participants weight – *upw*			$\sum$

Table 3. Weights of business diagrams categories

Business Diagram Type	Description	Weight
Simple	1-5 activities or 3-11 communications (communications + transactions)	5
Average	6-10 activities or 12-18 communications (communications + transactions)	10
Complex	11 and more activities or 18 and more communications (communications + transactions)	15

Table 4. Total unadjusted business diagram weight

Business Diagram Type	Activities Number	Business Diagram Weight	Number of Business Diagrams	Total
Simple	1-5	5	s	$5 \times s$
Average	6-10	10	a	$10 \times a$
Complex	11 and more	15	p	$15 \times p$
Total unadjusted business diagram weight – *ubdw*				$\sum$

The business diagrams are divided into the following three categories:

simple–average–complex.

Separation is performed on the base of activities number and transactions number. Each level of complexity receives the weight according to number of activities and communications (Tab. 3). Next, the number of business diagrams in individual categories is counted and it is multiplied by the assigned weight. The rows are summed to get the final result (Tab. 4).

Unadjusted BORMp (uBORMp). Total unadjusted BORMp is the sum of the unadjusted participant weight (upw) and the unadjusted business diagram weight ($ubdw$) described above:

$$uBORMp = upw + ubdw.$$

2.3 Adjusted BORMp Part

The adjusted part of BORMp revises the exact result provided by the unadjusted part. Adjusted factors are selected so that they should cover all influences that may have impact on the complexity of the developed IS.

The adjusted part consists of three factors. Two of them come from UCP and one factor is newly introduced. It was designed based on UCP analysis and identification of its weaknesses.

The technical factor is completely shared with UCP and the environmental factor is partly modified. The new factor is a so-called *customer factor*. The customer factor takes into consideration influence of the customer on the project complexity. The customers usually have some sort of influence on the IS complexity and the aim of customer factor is to reflect this influence.

The BORMp is an original method, whose main developer is one of the paper authors. The details about the used approaches for setting up factor weights and coefficients could be found in [10].

The following paragraphs introduce individual factors, their components, sub-factors, developed weights and coefficients.

The Technical Factor. The technical factor consists of 13 sub-factors that specify technical aspects of designed information system. The evaluation scale is 0 to 5. A sub-factor with no influence gets 0, the most crucial sub-factor gets 5 (see Tab. 5).

Table 5. Technical factor

Factor	Description	Weight	Calculation	=	Total
t_1	Distributed system	2.0	$t_1 \times 2.0$	=	
t_2	Response time or throughput performance objectives	1.0	$t_2 \times 1.0$	=	
t_3	End user efficiency	1.0	$t_3 \times 1.0$	=	
t_4	Complex internal processing	1.0	$t_4 \times 1.0$	=	
t_5	Code must be reusable	1.0	$t_5 \times 1.0$	=	
t_6	Easy to install	0.5	$t_6 \times 0.5$	=	
t_7	Easy to use	0.5	$t_7 \times 0.5$	=	
t_8	Portable	2.0	$t_8 \times 2.0$	=	
t_9	Easy to change	1.0	$t_9 \times 1.0$	=	
t_{10}	Concurrent	1.0	$t_{10} \times 1.0$	=	
t_{11}	Includes special security objectives	1.0	$t_{11} \times 1.0$	=	
t_{12}	Provides direct access for third parties	1.0	$t_{12} \times 1.0$	=	
t_{13}	Special user training facilities are required	1.0	$t_{13} \times 1.0$	=	
Total technical factor – *tFactor*					$\sum$

It is important to carefully assign suitable weights to individual sub-factors according to their expected impact on the designed IS. Definitions of the introduced sub-factors are follow:

Distributed system – complexity of distributed IS architecture (if IS runs on more computers),
Response time or throughput performance objectives – performance requirements factors,
End user efficiency – how effienciently must the system be fine-tuned for the user interaction,
Complex internal processing – complexity of external processing,
Code must be reusable – usability of the whole or part of code for programming of other applications or their parts,
Easy to install – the system installation complexity,
Easy to use – users' learnability and understandability,
Portable – IS portability to other platforms and environments,
Easy to change – complexity level of the changes of the system,
Concurrent – aspects of possible parallel development of individual system parts,
Includes special security objectives – alignment to special security requirements,
Provides direct access for third parties – level of third parties access support and related aspects,
Special user training facilities are required – influence of special training needs.

BORMp introduces the same sub-factors of the technical factor as UCP. These sub-factors cover well the technical area of software development.

The sub-factors with high impact on the project should be identified and evaluated with the highest weight. The assigned values (0-5) are multiplied with the sub-factor weights and then summed. The technical factor ($tFactor$) is computed this way, next it is used in the following formula – the technical complexity factor (tcf):

$$tcf = 0.4 + (0.03 \times tFactor).$$

The Environmental Factor. The environment is in BORMp understood from the view of supplier and that is the reason why supplier's employee skills, used equipments or methods are evaluated in the software development project. These influences are covered by the environmental factor. The evaluation is the same as for the technical factor. So firstly 7 sub-factors are evaluated by weights (0 – no influence, 5 – the most considerable influence). Their summary is in Tab. 6.

Definitions of introduced sub-factors:

Familiar with the development process – level of employee knowledge of the methodology used for project development,
Application experience – level of employee knowledge of tools that are used for IS development (for modelling, developing, etc.),
Object-oriented experience – employee knowledge of object-oriented environment,
Lead analyst capability – capability of the lead analyst including the capacity that he can dedicate to the estimated project,
Motivation – level of motivation to complete the project in time and in costs,
Part-time staff – number of part-time employees working on the project,
Difficult programming language – complexity of programming language(s).

Table 6. Environmental factor

Factor	Description	Weight	Calculation = Total
e_1	Familiar with the development process	1.5	$e_1 \times 1.5$ =
e_2	Application experience	0.5	$e_2 \times 0.5$ =
e_3	Object-oriented experience	1.0	$e_3 \times 1.0$ =
e_4	Lead analyst capability	0.5	$e_4 \times 0.5$ =
e_5	Motivation	1.0	$e_5 \times 1.0$ =
e_6	Part-time staff	1.0	$e_6 \times 1.0$ =
e_7	Difficult programming language	1.0	$e_7 \times 1.0$ =
Total environmental factor – *eFactor*			$\sum$

The procedure is the same as in the case of the technical factor. The evaluated sub-factors with assigned values are multiplied with sub-factors weights and then summed to give the total environment factor ($eFactor$). This factor is used in the environmental complexity factor (ecf).

$$ecf = 1.7 + (-0,015 \times eFactor)$$

Customer Factor. Customer factor is a new view on the software development process incorporated in the BORMp estimation. Its aim is to reflect the impact of customer requirements to the project. As was mentioned above, uncoordinated customer requirements may significantly affect efforts of information system development.

Procedure of the factor evaluation is the same as of the technical and the environmental factors. Seven customer sub-factors are evaluated by the weights 0-5 (0 – no influence, 5 – the most considerable influence). The factors are shown in Tab. 7.

Sub-factors' definitions:

Knowledge of IS – level of IT knowledge of customer's project team,
Customer's project manager capacity – capacity which the customer's project manager could dedicate to the project,
Customer's project members capacity – capacity which the project team members could dedicate to the project,
Knowledge of project organisation – understanding of the project organization and time schedule,
Connection with existing IT projects – relation to other IS projects,
Complexity of replaced IS – in the case that the new system should replace an old one,
Balance of requirements – customer ability to identify and communicate requirements and their expected changes.

The subfactors values are multiplied by their weights and then summed. Theresult is the total customer factor ($cFactor$) which is used to determine customer complexity factor (ccf):

$$ccf = 0,5 + (0,01 \times cFactor).$$

Table 7. Customer factor

Factor	Description	Weight	Calculation = Total
c_1	Knowledge of IS	0.5	$c_1 \times 0.5$ =
c_2	Customer's project manager capacity	2.0	$c_2 \times 2.0$ =
c_3	Customer's project members capacity	1.5	$c_3 \times 1.5$ =
c_4	Knowledge of project organisation	0.5	$c_4 \times 0.5$ =
c_5	Connection with existing IT project	2.0	$c_5 \times 2.0$ =
c_6	Complexity of replaced IS	1.5	$c_6 \times 1.5$ =
c_7	Balance of requirements	1.0	$c_7 \times 1.0$ =
Total customer factor – *cFactor*			$\sum$

2.4 The Productivity Factor

The productivity factor is an important input for methods of complexity estimation. It is a recommended number of man-hours per one BORM point in dependence on various influences (e.g. experience of project team, size of the developed IS, etc.).

The value of the productivity factor is set to 30 man-hours per one BORMp according to expert estimation. Because of the customer factor, the value is set higher than for UCP (20 man-hours per use-case).

3 Total BORMp

The above computed values are connected by one formula that counts the result of the adjusted BORMp. The formula consists of the unadjusted part (the participant and the business diagram numbers) and the technical, the environmental and the customer factor:

$$aBORMp = uBORMp \times tcf \times ecf \times ccf.$$

The complexity is now specified by a non-dimensional number which is the result of the $aBORMp$ formula. To get the actual effort it is necessary to multiply the adjusted BORMp by the productivity factor:

$$Effort = aBORMp \times pf.$$

4 BORMp Testing

The verification of described BORMp method was performed on the projects of one international consulting company. It is hard to get to actual data, because companies usually classify their internal data as sensitive and to be used outside. That's the reason why the tested projects are described just in general.

4.1 Projects Description and Testing Procedure

Project 1 – Project of information system (IS) development was delivered for an important Czech logistic company. The project could be evaluated as complex. The project costs were apx. 20 mio. CZK (8 mio. EUR). The project team consisted of marketing, procedural and informatic groups.

Project 2 – Project of internal IS development which cost apx. 1 mio. CZK (40 ths. EUR). The project was classified as small.

The method was applied step by step according to the described procedure above.

4.2 Results

The table 8 shows that the effort estimation of project 1 is lower than its real effort. On the other hand the estimation effort of project 2 provided higher value. There are complex reasons for this, but the conclusion is that the method provides under-valued estimations for complex projects and over-valued estimations for small projects. These findings will be taken into consideration during further testing.

Table 8. BORMp Testing Results

		Project1	Project2
Unadjusted Part	Unadjusted participant weights	30.0	22.0
	Unadjusted business diagram weights	350.0	270.0
	Unadjusted BORMp	380.0	292.0
Adjusted Part	Technical factor	1.3	1.1
	Environmental factor	1.4	1.5
	Customer factor	0.8	0.7
Total BORMp		531.8	316.0
Productivity factor (man-hours /1 aBORMp)		30	
Total estimated effort (man-hours)		**15 955**	**9 480**
Total real effort (man-hours)		**17 032**	**7 040**

Results Evaluation. The final estimation accuracy for project 1 is greater than 93%, which is very sufficient for engineering use. The estimation accuracy for project 2 is just 65%.

5 Conclusions

The paper introduces the BORMp method, which has been developed and tested for a few years. The results of tests made so far show a potential for good estimations for complex projects. The testing will continue to reach some statistically more adequate

number of tests. Further research will be focused on improving the accuracy for various types of projects and other possible utilisation: we plan trying to replace the productivity factor by COCOMO. It means that BORMp would generate complexity estimation only and the effort would be estimated by COCOMO which would use BORMp result as its input.

The efforts are aimed also to introducing BORMp to a broader professional interest that could help to improve the method provide more feedback.

Acknowledgements. This paper was elaborated under the grant no. 2C06004 Information and Knowledge Management (IZMAN) of Ministry of Education of Czech Republic and the grant no. 200811140035 of Grant Agency of The Faculty of Economics and Management of the Czech Univerzity of Life Sciences in Prague.

References

1. Albrecht, A.J., Gaffney Jr., J.E.: Software Functions, Source Line sof Code and Development Effort Prediction.: A Software Science Validation. IEEE Transactions on Software Engineering (TSE) 9(6)
2. Carda, A., Merunka, V., Polák, J.: The art of system design (in czech), Grada, ISBN 80-247-0424-2
3. Hall, J., Merunka, V., Polák, J., et al.: Accounting information systems - Part 4: System development activities, Thomson South-Western New York, 4th edn., ISBN 0-324-19202-9
4. International Function Point Users Group: IT Measurement Practial Advice from the Experts. Addison-Wesley, Boston, ISBN 0-201-74158-X
5. Karner, G.: Use Case Points - Resource Estimation for Objectory Projects, Objective Systems SF AB (copyright owned by Rational Software)
6. Knott, R., Merunka, V., Polak, J.: The BORM methodology: A Third-generation fully object-oriented methodology. In: Knowledge-Based Systems. Elsevier Science International, New York, ISSN 0950-705
7. Liping, L., Roussev, B., Knott, R., Merunka, V., Polák, J., et al.: Management of the Object-Oriented Development Process - Part 15: BORM Methodology, pp. 1–59140. Idea Group Publishing, ISBN 1-59140-605-6
8. Merunka, V.: Object oriented database normalization. In: The proceeding conference Objects, Prague (2004) ISBN 80-248-0672-X
9. Merunka, V.: BORM – Overview of the methodology and case study of agrarian information system In: UZPI Agriculture economics, Prague, ISNN 0139-570X
10. Struska, Z., Brožek, J.: Approaches for setting up BORM points coefficients (in Czech). In: Proceeding conference of Agrarian perspectives (2007) ISBN 978-80-213-1675-1
11. Struska, Z., Merunka, V.: BORM points – New concept proposal of complexity estimation method. In: ICEIS 9ĥ International Conference on Enterprise Information Systems, Madeira Portugal, ISBN 978-972-8865-90-0
12. Struska, Z.: Complexity Estimation Method in Object Oriented Environment – Function and Feature points. In: The proceeding conference of Objekty, Ostrava (2005) ISBN 80-213-0682-3
13. Vaníček, J.: Measurement and estimation of information system quality (in Czech), PEF ČZU Prague, ISBN 80-213-0667-X

A Technology Classification Model for Mobile Content and Service Delivery Platforms

Antonio Ghezzi, Filippo Renga, and Raffaello Balocco

Politecnico di Milano, Department of Management, Economics and Industrial Engineering
Piazza Leonardo da Vinci 32, 20133 Milan, Italy
{Antonio1.Ghezzi,Filippo.Renga,Raffaello.Balocco}@polimi.it

Abstract. The growing complexity of mobile "rich media" digital contents and services requires the integration of next generation middleware platform within Mobile Network Operators and Service Providers infrastructural architecture, for supporting the overall process of content creation, management and delivery. The purpose of the research is to design a technology classification model for Content & Services Delivery Platforms – CSDPs –, the core of Mobile Middleware Technology Providers – MMTPs – value proposition. A three-steps theoretical framework is provided, which identifies a set of significant classification variables to support the platforms positioning analysis. Afterwards, through adopting the multiple case studies research methodology, the model is applied to map the current CSDP offer presented by a sample of 24 companies, classified as MMTP, so to test the framework validity and get a valuable insight on the actual "state of the art" for such solutions. The main findings show that existing platforms possess major strengths – e.g. wide content portfolio manageable, integration between mobile and web channels and frequent recourse to SOA and Web Service approach –, while some drawbacks – poor support to context aware and location-based services, verticality and low interoperability of some proprietary products, criticality of content adaptation etc. – are still limiting the solutions effectiveness.

Keywords: Mobile Communications, Mobile Content & Service Delivery Platform, Technology Classification Model, Multiple Case Studies.

1 Introduction

In a recent past of the Mobile Content market, when contents and services offered by Mobile Network Operators (MNOs) and the first Mobile Content & Service Providers (MCSPs) were quite simple – eg. SMS, monophonic ringtones etc. –, the administration activities were carried out through *ad hoc* "legacy" systems; delivery and billing of services were managed through operators' SMSCs (Short Message Service Centres). The need of integrated platforms for managing the value added services portfolio was not strongly felt about [1] [2].

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 600–614, 2009.
© Springer-Verlag Berlin Heidelberg 2009

However, the growing complexity and cost of mobile "rich media" digital contents, the rise of the off-portal environment, the problems of compatibility with different device models and the necessity of handling articulated billing models forced the MNOs to further develop their legacy systems, thus enhancing their functionalities. Nevertheless, such in-house developed "first generation" Service Delivery Platform proved themselves unable to face the emerging market needs [3] [4] [5].

Today, Mobile Content market has evolved to a degree of complexity not any more faceable though unsafe, non-flexible and non-scalable first generation system, and requires the introduction of "second generation" platforms. These solution, offered by Mobile Middleware Technology Providers (MMTPs), are here named "Mobile Content and Service Delivery Platforms" (CSDPs), and can be defined as middleware platforms combining a wide set of functionalities – consistently aggregated into different modules –, and equipped with network-side and device-side interfaces, thus creating an integrated suite with the purpose of supporting some or each phase of the mobile digital content creation, management & delivery process.

Unlike the previous solutions, next generation platforms possess the following characteristics: scalability and flexibility; adoption of open standard and of "best of breed" components; support to multiple relationships with developers and Content Providers (CP); capability of handling a large portfolio of contents and services demanded by a wide range of mobile devices; common and reusable interfaces with Business Support Systems and Operation Support Systems [5] [6] [7] [8].

The introduction of a CSDP within operators and service providers' IT architecture allows to obtain a wide set of benefits, as argued by a vast literature [4] [5] [6] [9] [10] [11] [12] [13] [14] [15].

Concerning the creation, management and delivery of contents and services, second generation platforms grant higher efficiency, higher control on the service lifecycle, lower development costs and shorter time to market; moreover, CSDP adoption enables the widening of service portfolio, thus leveraging on the "long tail theory" [16], and making possible to exploit scale and scope economies.

With regard to the operators and service providers technology infrastructure, CSDP grant some major advantages: the unification of service creation, execution and management environments; the integration of multiple delivery channels; a simplification of interfacing with third parties; an higher architectural flexibility and scalability; an increased interoperability with legacy systems; and an overall reduction of technological complexity.

Anyway, to grasp the previous benefits, CSDPs shall be designed according to specific technical concepts and approaches, and shall possess certain key characteristics, that will be later discussed.

Taking from a vast literature review, the purpose of this paper is to develop a reference model for classifying Mobile Content & Service Delivery Platforms, through the identification of a set of significant technology dimensions or classification variables. Afterwards, through adopting the multiple case studies research methodology, the model will be employed to map the current CSDP offer presented by a sample of 24 companies, classified as MMTP, so to test the framework validity and get a valuable insight on the actual "state of the art" for such solutions.

2 Methodology

The development of the original CSDP technology classification model followed three main steps.

At first, taking from a wide literature analysis, a thorough functional architecture of CSDP was created. Such architecture will be employed to identify a set of platform categories, in terms of their main objectives – depending on the range of functionalities possessed –.

The second step requested the identification of noteworthy technology dimensions or classification variables, whose relevance was determined from their impact on the achievable benefits.

The third and last step is based on the creation of a set of "classification matrices" – emerging from the combination of previously identified technology dimensions – to support the mapping of the platform solutions offered by MMTPs.

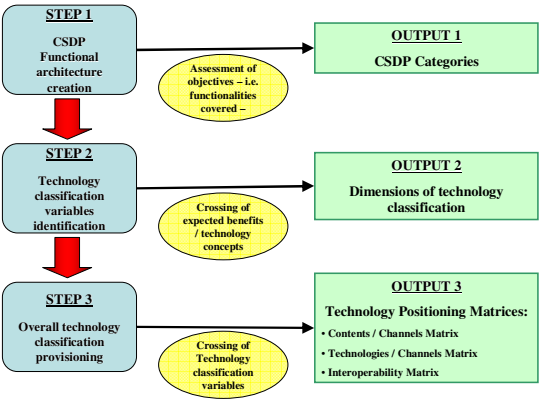


Fig. 1. The CSDP technology classification model main steps and outputs

As the technology classification model is developed with the purpose of supporting the classification of CSDPs currently offered by MMTPs, the model will hence be applied to a sample of platforms present in the market.

To collect both qualitative and quantitative information concerning MMTPs' products and solutions, as well as the overall value propositions presented by such player typology, the multiple case studies research methodology was employed [17]: From January to July, 2008, 24 in-depth exploratory case studies – based on 72 both face-to-face and phone semi-structured interviews – on Mobile Middleware Technology Providers were performed, focusing on the set of variables and dimensions identified through the literature analysis. Coherently to the research methodology employed [18], firm sample was not randomly selected, but firms were picked as they conformed to the main requirement of the study, while representing both similarities and differences considered relevant for the data analysis. The main predetermined filters used to discriminate among firms were: the presence of a well-defined line of business – if not the core business – dedicated to the commercialization of Content and Service Delivery

Table 1. Theoretical sample of companies interviewed

Sample of companies	
Alcatel-Lucent	Microsoft
Bea Systems	Nec
Beeweeb	Neodata
Comverse	Nokia-Siemens Networks
Dylogic	Openwave
Ericsson	Polarix
Fabbrica Digitale	Qualcomm
First Hop	Reitek
HP	Reply
IBM	Sybase 365
LogicaCMG/Acision	Txt Polymedia
Mblox	Xiam Technologies

Platforms or CSDP modules; and the presence of an offer directed to the Mobile Telecommunications market. The following table provides the full list of analyzed companies.

A multiple case study approach reinforced the generalizability of results [19], and allowed to perform a cross analysis on platform characteristics and their combinations – to see which variables changed and which remained constant –, due to the presence of extreme cases, polar types or niche situations within the theoretical sample [19].

As the validity and reliability of case studies rest heavily on the correctness of the information provided by the interviewees and can be assured by using multiple sources or "looking at data in multiple ways" [17] [20], multiple sources of evidences or research methods were employed: interviews – to be considered the primary data source –, analysis of internal documents, study of secondary sources – research reports, websites, newsletters, white papers, databases, international conferences proceedings –. This combination of sources allowed to obtain "data triangulation", essential for assuring rigorous results in qualitative research [21].

3 The CSDP Technology Classification Model

3.1 The CSDP Functional Architecture

The integrated assessment of an academic literature focusing on middleware platforms, of technical documents elaborated by MMTPs and of market research reports made possible to build an architectural reference model for a CSDP.

Such model identifies 7 modules, in turn entangling 48 functionalities or sub-modules, which grant the platform operation. In addition, 3 cross-modules macro-functions are evidenced. The 7 modules can be briefly described as follows.

1. "Third party management" module supports the relationships with third parties – MCSPs and CPs – cooperating with the platform owner.
2. "Business management" module encompasses the functionalities related to managing the activities of mobile digital contents & services business as a whole.

3. “Content management” module is dedicated to the end to end handling of digital contents published on the platform.
4. “Service management” module deals with the management of value added services – comprising a combination of digital contents and/or other applications – offered.
5. “Content adaptation” module performs the functionalities of contents/services adaptation according to user profiles, device profiles and network capabilities.
6. “Subscriber management” module supports the process of managing the customer base, containing information on preferences and device potentialities – representing the key inputs of content adaptation activities –.
7. “Network adaptation” module handles the different activation channels and the main interfaces to different access networks.

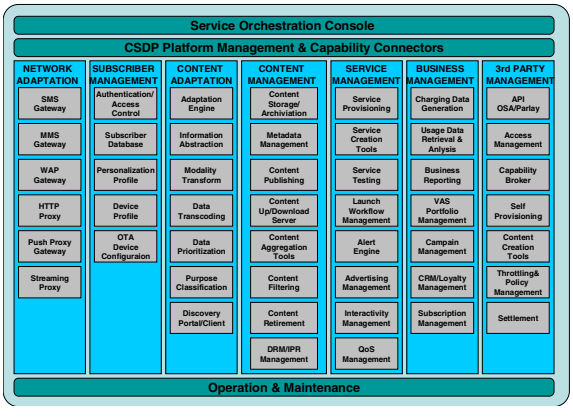


Fig. 2. The CSDP functional architecture

The 3 cross-modules functions enable the creation of am integrated and common environment.

1. “Service orchestration console”, leveraging on the concepts of Service Oriented Architecture (SOA) , Web Services and IP Multimedia Subsystem (IMS), allows to enhance the efficiency and effectiveness of service management, through the reuse of service components and applications etc.
2. “Platform management & capabilities connectors” handles the processes of service creation, execution and management, and coordinates the interconnections between the different modules, thus working as an integration layer of the platform’s core functionalities.
3. “Operations & maintenance” supports the platform operativeness and maintenance.

Within the overall technology classification model, the above described functional architecture will serve as a first tool for providing a CSDP classification in terms of modules and functionalities covered by the existing platforms, giving rise to a list of different platform categories.

3.2 The Identification of the Model's Technology Classification Variables

The identification of key technology variables or dimensions is essential to provide the basis for the CSDP classification and benchmarking process, as it makes possible to discriminate the technical origins of different platform performances.

The *rationale* followed to judge a dimension's significance was its impact on the achievable benefits. A technology concept is relevant if its presence or absence influences, to some or to a large extent, the attainability of expected benefits deriving from the CSDP introduction, constituting a plus or a drawback for the platform itself.

Taking into account the adoption benefits pinpointed in section 1, and leveraging on a wide technical literature, 12 key technology dimensions were identified as strongly impacting on the expected benefits from CSDP introduction.

1. *Delivery channels available*. Through the "Network adaptation" modules and network elements like Media Gateway, Media Switch and Media Router [22] [23], CSDP are able to deliver contents on multiple channels: fixed, mobile, mobile broadcast (DVBH), IP, wireless, satellite, digital terrestrial tv. The availability of a wide range of delivery channels allows to reach a larger customer base, thus increasing the contents selling revenues; moreover, it reduces technology complexity thanks to the unified delivery environment, and makes possible to exploit scale and scope economies in distribution.
2. *Content types treated*. The main reach media digital contents the platform can manage are: mobile games; video; music; infotainment – microbrowsing, SMS, MMS –; Personalization – logos, wallpapers, ringtones, ring-back tones – [24]. The platform capability of treating the lifecycle of different content types impacts on several benefits, like the enhancement of services management efficacy and efficiency, the widening of service portfolio and potential customers and the increase of revenues coming from Mobile Content.
3. *Media types and formats supported*. Strictly related to the "content types" variable, this dimension assumes great relevance because of the growing multimediality of contents, embedding audio, video, images, graphics and messaging [25]. Though the support to multiple media types and formats increases the platform complexity, it positively impacts on the width of the contents & services portfolio.
4. *Proprietary vs. Open Source technology employed*. The trade-off here presented is between vertical, end to end proprietary platforms and open standards-based solutions. While the former option is related unique products, hardly replicable by competitors and potentially generating lock-in effects as regard to customers – MNOs and MCSPs –, the latter option makes the platform more flexible and easily interoperable with legacy and third parties systems [26].
5. *Service Oriented Architecture and Web Services adoption*. The introduction of SOA allows to depart from a point-style approach in platform design, ensuring a full connection between BSS/OSS and the platform itself, also allowing the integration of different applications and the reusability of service components, through transversal orchestration functions [7] [27] [28]. In addition to this, Web Services grant interoperability between distributed applicative components, representing a service layer the SOA leverages on to access to

different contents and services and to combine them so to create new applications [4] [15] [29]. Therefore, the adoption of a pervasive SOA and Web services approach impacts on several potential benefits: the increase of efficiency and automation of value added services (VAS) lifecycle management; the services time to market reduction; the widening of offer portfolio; the ability of exploiting scale and scope economies; the reduction of technology complexity, and the architectural flexibility and scalability.

6. *IP Multimedia Subsystem adoption*. IMS can constitute the standard on which to create dedicated architectures for IP multimedia services distribution to end customers [30]. The IMS key concepts are close to those proposed by the SOA and Web Services approach [15], pushing towards the reuse of applicative components and the creation of a common "control layer" to centralize the management of services published on the CSDP. This increases efficacy and efficiency in the VAS portfolio management, making the architectural solution more flexible and scalable.
7. *OSA/Parlay Interfaces integration*. OSA/Parlay Application Program Interfaces offer an abstraction of core network functionalities, supporting the interfacing between the platform and third parties systems [31] [32]. [33] [34]. Specifically, API Parlay X leverage on Web Services technologies, letting the emergent developers community to easily access network functions and capabilities [35]. Therefore, API influence new services' time to market reduction and the simplification of the relationships with business partners.
8. *Interactivity and two-way channels availability*. Making interactivity and two-way communication available to end users can increase the service perceived "quality of experience", also allowing the appealing upload of "user generated contents" on the platform – thus making the end customer become a content provider of his own –. This can differentiate the platform from competitors' offers.
9. *Context aware & location-based services enablement*. The possibility of delivering forefront services based on the context of fruition – determined by network capabilities, device profile and user profile – and on the end user geo-spatial location rests on the platform equipment of technologies for "network discovery", user & device profiles storing and GPS localization. This all enhances the innovativity of the offer, with a potential positive influence on revenues generated – depending on the services uptake –.
10. *Out of the box vs. taylor made solution*. As Porter [36] asserted, strongly standardized and poorly customizable products bring down both the technology complexity and the offer differentiability; on the contrary, taylor made solutions imply higher development costs, but grant offer uniqueness.
11. *Application Development Platforms supported*. Concerning software technologies allowing the creation and consequent fruition of mobile applications – Sun Microsystems' J2ME, Qualcomm's Brew, Macromedia's Flash Lite, W3C's SVGT, Streamezzo's Laser etc. [37] –, it can be argued that supporting a wide range of ADPs positively impacts on the range of contents deliverable; however, the proprietary nature of some ADP solutions can make interoperability and third parties relationships more complex.

12. *Mark-up languages supported*. Within the CSDPs perimeter, mark-up programming languages belonging to the HTML family [38] – XML, XHTML, XSL, SGML, WSDL, PML, SMIL, VXML, SALT, SAML – have two main purposes: first, they represent the codes used for platform development, and ensure the overall technology infrastructure governability; second, they support the creation of multimedia applications. Relying on such languages increases the efficiency in the VAS lifecycle management, making the architectural solution more flexible, scalable and interoperable.

The previously identified technology variables are meant to be used to classify the existing CSDP solutions, as shown in the next section.

4 The Model Application to the Companies Sample

In order to accomplish the second major research objective, i.e. to provide a classification of the current CSDP offer presented by MMTPs, the theoretical model developed will be applied to a real context, represented by the sample of firms analyzed through case studies.

The functional architecture will serve to define the main CSDP categories, in terms of platform purposes – implied by the set of functionalities available –.

The so identified categories will be then mapped through a set of classification matrices, which combine the 12 technology dimensions, thus providing a visual representation of real world platform clusters.

4.1 The CSDP Categories

As stated earlier, the first step of the CSDP technology classification model seeks to employ the CSDP functional architecture to discriminate between the platforms offered by the MMTPs under scrutiny in terms of their main purposes, starting from the assumption that such purposes can be inferred from an evaluation of the modules and functionalities covered – which, in fact, enable the execution of the platforms tasks.

According to the key functionalities offered, it was possible to identify 5 distinct CSDP categories, characterized by different purposes.

1. *Content Creation platforms*. These CSDPs' main functionalities are related to the activities of concept, development and production of the digital content or service. They offer tools for service creation, workflow management, service testing, as well as for aggregation of internally produced and third parties uploaded contents. Within the theoretical sample of companies analyzed through case studies, only 1 platform was classified as focused on content creation.
2. *Content Management platforms*. Such platforms mainly cover the activities spanning from content publishing to content delivery, offering several functionalities: content storage, publishing, aggregation, filtering, retirement; metadata management; digital rights and intellectual propriety rights management; content adaptation; authentication and access control; user &

device profiles management; over-the-air configuration; third parties relationship management. In the companies sample, 8 players offered content management platforms.

3. *Business Management platforms*. The platforms are meant to handle digital contents in a wider business perspective, ensuring the integration between the specific VAS business and legacy systems – e.g. BSS/OSS, database and data warehouse, Customer Relationship Management, Enterprise Resource Planning, billing & accounting system. The key functionalities are related to service orchestration, reporting, portfolio and campaigns management, subscribers management. In the sample, 7 solutions could be labelled as business management platforms.
4. *Transactional platforms*. These solutions are interconnected to MNOs' systems, and support the activities related to the so called "CBA process": content charging, content billing, and revenues accounting among the involved parties. These CSDPs commonly possess some functionalities of SMS/MMS/WAP-based service delivery. Only 2 solutions provided by the MMTPs under consideration are transactional platforms.
5. *Transversal platforms*. Such CSDPs show transversal coverage of modules and functionalities, that makes difficult to identify a prevalent purpose. The category is populated by 6 solutions.

The 5 homogeneous CSDP categories will be further analyzed according to the 12 technology dimensions, so to obtain a deeper insight on the platforms characteristics.

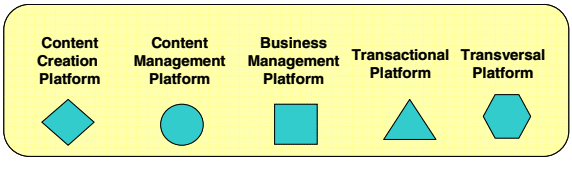


Fig. 3. The five CSDP categories

Figure 3 shows the symbols employed in the overall technology provisioning matrices to identify each platform category.

4.2 The CSDP Classification Matrices

The last step of the classification model consists in creating a set of original positioning matrices, so to provide an overall technology classification of CSDPs. Through such analysis, it will be possible to support a technical benchmarking between the solutions currently available on the market, pinning down and interpreting their main positive and negative elements, and in the end drawing insightful conclusions on the offer state of the art.

The multidimensional matrixes are created through the crossing of the 5 CSDP categories and the 12 classification variables.

The first six-dimensional matrix, called "Contents/Channels Matrix", considers the following variables: CSDP category; delivery channel – ranging from monochannel,

i.e. mobile, or multichannel –; content portfolio supported – ranging from transversal, i.e. supporting many content types, to focalized support; Formats support, which can be narrow or wide, with reference to the number of media types and formats handled; context aware and location-based services availability; presence of interactivity and two-ways channels.

By observing the CSDPs positioning, the prevalence of platforms capable of delivering a wide range of contents clearly emerges. Moreover, the multichannel option is followed exclusively by platforms offering a large content portfolio: this finding can be explained by considering that the investments required for the integration of different delivery are only justifiable if high revenues coming from a wide VAS offer are expected. The interactivity feature is also widespread in the sample, being present on 19 platforms out of 24.

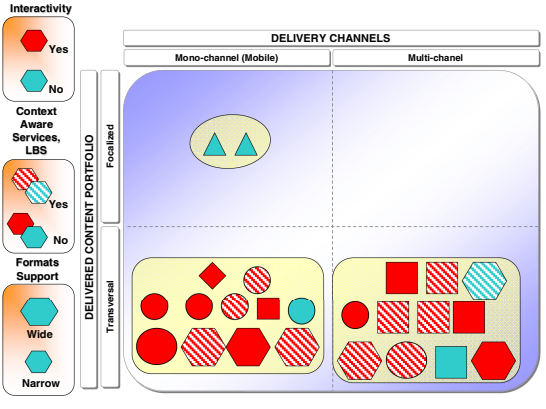


Fig. 4. Contents/Channels matrix

This first mapping gives rise to 3 different CSDP clusters:

1. Monochannel / Focalized content portfolio platforms – 2 solutions –;
2. Monochannel / Transversal content portfolio platforms – 11 solutions –.
3. Multichannel / Transversal content portfolio platforms – 11 solutions –.

The second five-dimensional matrix created, called "Technologies/Channels Matrix" classifies the platforms in terms of: CSDP category; delivery channel; technology employed, distinguishing between proprietary and open source technology; SOA adoption; IMS adoption.

The variables crossing appears significant, as it explicitly addresses the correlation between open technologies, multiple channels, SOA and IMS.

The SOA approach is adopted in 16 products out of 24, demonstrating the validity of this architectural paradigm coming from the IT enterprise platforms environment, and quickly diffusing in the Telecommunications context. IMS is employed in the majority of products, testifying the service layer evolutions towards an "all IP" approach. Proprietary technologies are preferred to open source ones, as MMTPs struggle to make their offers unique, and potentially lock-in their business customers.

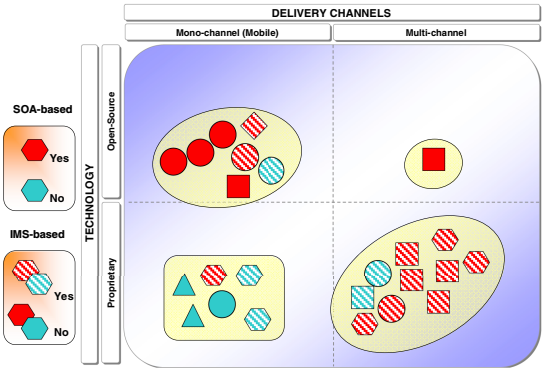


Fig. 5. Technologies/Channels matrix

The products collocation in the map shows that while SOA approach is more common along the open source axis, the IMS adoption is frequent in the multichannel alternatives, regardless of the "technology" variable, in the light of the growing significance of IP in the process of integrating different delivery technologies.

From the second map, 4 clusters emerge:

1. Proprietary technology / Monochannel platforms – 6 solutions;
2. Open source technology / Monochannel platforms – 7 solutions.
3. Proprietary technology / Multichannel platforms – 10 solutions
4. Open source technology / Multichannel platforms – 1 solution.

The third six-dimensional classification map, called "Interoperability Matrix", builds up a relationship between the following variables: CSDP category; technology employed; customizability level, which can assume the values of "out of the box" standard solutions, tailor made customized solutions, or standard + custom – where both alternatives are possible –; OSA/Parlay API availability; mark-up languages support; ADP support, ranging from narrow to wide support.

The map is created with the purpose of shedding light on the cross effect of technology variables impacting on the platforms' interoperability with legacy or third parties systems.

Analyzing the CSDP positioning, a consistent fragmentation becomes evident. APIs are widely used, as well as mark-up languages; on the other hand, the support to different ADPs is still narrow, because of some "standard wars" between proprietary technologies the regulators or the international consortiums will be asked to settle.

Specifically, 6 different clusters can be individuated:

1. Proprietary technologies / out of the box platforms – 6 solutions –;
2. Proprietary technologies / tailor made platforms – 4 solutions –;
3. Proprietary technologies / standard or custom platforms – 6 solutions –;
4. Open source technologies / out of the box platforms – 3 solutions –;
5. Open source technologies / tailor made platforms – 1 solution –;
6. Open source technologies / standard or custom platforms – 4 solutions –.

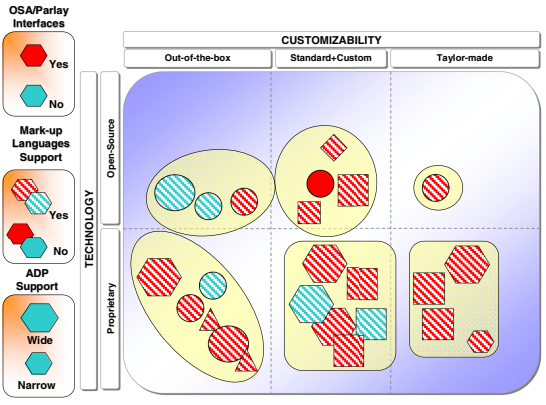


Fig. 6. Interoperability matrix

4.3 The Offer State of the Art

The picture obtained through the set of technology classification matrices allows to make insightful inferences on MMTPs offer state of the art, in terms of both strengths and weaknesses.

The main pluses characterizing the offer can be synthesized as follows:

- wide portfolio of contents and services deliverable;
- widespread support to interactivity;
- integration between mobile and web channels;
- wide support to media types and formats;
- frequent SOA and IMS adoption;
- significant modularity, flexibility and scalability;
- frequent OSA/Parley API adoption;
- common use of mark-up languages.

On the other side of the coin, the current CSDP offer is characterized by some significant drawbacks:

- scarce support to context aware and location-based services;
- verticality and poor interoperability of some proprietary products;
- criticality of content adaptation processes;
- low products customizability;
- limited horizontal support to ADP.

5 Conclusions

The research provided an original reference model for supporting a technology classification of mobile Content & Service Delivery Platforms.

Such original framework, well grounded on existing literature, was hence applied to a sample of platforms currently marketed by 24 Mobile Middleware Technology Providers, so to test its validity, and obtain a valuable insight on the CSDP offer state

of the art. The findings show the real world offer of middleware platforms possesses some interesting features – ranging from the width of service offered and delivery channels supported, to the adoption of SOA and IMS approaches –; nevertheless, other significant drawbacks – e.g. insufficient support to context aware and location-based services, poor coverage to application development platforms etc. – are limiting the solutions effectiveness. Short term market trends will most likely see the coexistence of end to end transversal platforms, and of niche solutions focused on few modules or functionalities.

Concerning the model's properties, internal validity is ensured, for the platform positioning – dependent variable – is perfectly explained by the identified dimensions of classification – independent variables –; in terms of external validity, the model can be generalized to different populations, thanks to the width and significance of the sample under scrutiny; moreover, the rigorous qualitative research methodology employed grants the reliability and replicability of the model's results.

The paper's value for researchers can be brought back to the creation of a reference framework capable of modelling the emergent phenomenon related to the rise of middleware platform providers within the Mobile Content market.

The value for practitioners lies in the provisioning of a tool for mapping existing and future CSDP offer, establishing strong ties between platform capabilities and associated benefits, thus supporting the decision making process of a wide set of stakeholders – not only client firms attempting to find out what they should look for in middleware solutions and what they should adopt according to their needs, but also platform vendors themselves, to guide their offer positioning –.

Though representing a significant step towards the study of MMTPs through the evaluation of the core element of their value proposition, the research does not specifically assess the strategic and competitive implications of a given CSDP technology positioning for the platform provider. Future research will need to focus on integrating the present model within a thorough strategy analysis framework for middleware platform providers.

References

1. ABI Research: Mobile Content Delivery Platforms Enable Revenue Growth for Video, Games and music (2006a), http://www.abiresearch.com
2. ABI Research: Next Generation Service Delivery Platforms (2006b), http://www.abiresearch.com
3. Karlich, S., et al.: A self-adaptive service provisioning framework for 3G+/4G mobile applications. IEEE Wireless Communications (2004)
4. Pavlovsky, C.J., Staes-Polet, Q.: Digital Media and Entertainment Service Delivery Platform. IBM MSC (2005)
5. Forrester Research: Service Delivery Platform success requires a strategic vision and corporate collaboration. Trends (2007)
6. Ericsson: Service Delivery Platforms: efficient deployment of services. White paper (2006)
7. The Insight Research Corporation: IMS, SIP and Service Delivery Platforms: Telecom adoption of SOA and Enterprise applications 2007-2001. Report (2007)

8. iSupply Corp: Mobile Content Enablement Platforms: Software Platforms Monetize and Deliver Mobile Music, Games and Video (2007), http://www.isuppli.com
9. Sabat, H.K.: The evolving mobile wireless value chain and market structure. Telecommunications Policy (26), 505–535 (2002)
10. Hewlett-Packard: HP Service Delivery Platform. White paper (2005)
11. Brynjolfsson, E., Hu, Y., Smith, M.D.: From Niches to Riches: Anatomy of the Long Tail. Mit Sloan Management Review (2006)
12. Kuo, Y., Yu, C.: 3G Telecommunication operators' challenges and roles: a perspective of mobile commerce value chain. Technovation, pp. 1347–1356 (2006)
13. Noordman, M.: Squeezing the guy in the middle. Ericsson Business Review (24), 26–29 (2006)
14. Peppard, J., Rylander, A.: From Value Chain to Value Network: an Insight for Mobile Operators. European Management Journal 2(24) (2006)
15. Sur, A., Skidmore, D., Chakravarty, S.: Web Services based SOA for Next Generation Telecom Networks. In: IEEE International Conference on Service Computing (2006)
16. Anderson, C.: The Long Tail. Wired (2004)
17. Yin, R.: Case study research: Design and methods. Sage Publishing, Thousand Oaks (2003)
18. Pettigrew, A.: The management of strategic change. Blackwell, Oxford (1988)
19. Meredith, J.: Building operations management theory through case and field research. Journal of Operations Management (16), 441–454 (1998)
20. Eisenhardt, K.M.: Building theories from case study research. Academy of Management Review 4(14), 532–550 (1989)
21. Bonoma, T.V.: Case research in marketing: opportunities, problems, and a process. Journal of Marketing Research 22, 199–208 (1985)
22. Li, J., Zhang, J., Verma, S., Ramaswamy, K.: Mobile Content Delivery Through Heterogeneous Access Networks. IEEE, Los Alamitos (2003)
23. Aioffi, W., Almeida, J., Mateus, G., Mendes, D.: Mobile Dynamic Content Distribution Networks. MSWiM, Venezia (2004)
24. Bertelè, U., Rangone, A., Renga, F.: Il Mobile diventa Web. Il Web diventa Mobile. Osservatorio Mobile Content report (2008)
25. Mobile Entertainment Industry and Culture: WP4 – Mobile Technologies Deliverable D4.1.1: Existing and imminent Mobile Entertainment Technologies. IST-2001-38846 (2003)
26. Blind, K.: Interoperability of software: demand and solutions. In: Panetto, H. (ed.) Interoperability of Enterprise Software and Applications, pp. 199–210. Hermes Science, London (2005)
27. Gartner Research: Simplify Your Business Processes With an SOA Approach. Research report (2002)
28. Forrester Research: Real-world SOA: SOA Platform case studies. Tech. Choices (2005)
29. Capp, M., Farley, P.: Mobile Web Services. BT Technology Journal 23(2), 202–213 (2005)
30. 3rd Generation Partnership Project (3GPP): IP Multimedia Subsystem (IMS). TS23.228, Stage 2 (2006)
31. Zahariadis, T., et al.: Global Roaming in Next-Generation Networks. IEEE Communications Magazine 40(2), 145–151 (2002)
32. Moerdijk, A.-J., Klostermann, L.: Opening the Networks with Parlay/OSA: Standards and Aspects Behind the APIs. IEEE Network (3), 58–64 (2003)

33. European Telecommunications Standards Institute (ESTI) Standard: Open Service Access; Application Programming Interface (API); Part 1: Overview (Parlay 5)", ES 203 915-1 v1.11 (2005a)
34. Karlich, S.: An approach to mobile service delivery platforms. IST OPIUM Project, blue paper (2007)
35. European Telecommunications Standards Institute (ESTI) Standard: Open Service Access (OSA); Parlay X Web Services; Part 1: Common, ES 202 391-1 v1.11 (2005b)
36. Porter, M.: Strategy and the Internet. Harvard Business Review, 62–78 (2001)
37. Barsook, J., Freedman, E.: Mobile Content Delivery Technologies. IEEE, Los Alamitos (2005)
38. IETF: Hiper Text Markup Language (1995)
39. Balocco, R., Bonometti, G., Ghezzi, A., Renga, F.: Mobile Payment Applications: an Exploratory Analysis of the Italian Diffusion Process. In: ICMB 2008 (2008)

Patterns for Modeling and Composing Workflows from Grid Services

Yousra Bendaly Hlaoui and Leila Jemni Ben Ayed

Research Laboratory in Technologies of Information and Communication (UTIC)
Institute of Sciences and Techniques of Tunis, Avenue Taha Hussein, Tunis, Tunisia
yousra.bendalyhlaoui@esstt.rnu.tn, leila.jemni@fsgt.rnu.tn

Abstract. We propose a set of composition patterns based on UML activity diagrams that support the different forms of matching and integrating Grid service operations in a workflow. The workflows are built on an abstract level using UML activity diagram language and following an MDA composition approach. In addition, we propose a Domain Specific Language (DSL) which extends the UML activity diagram notation allowing a systematic composition of workflows and containing appropriate data to describe a Grid service. These data are useful for the execution of the resulting workflow.

Keywords: UML-Activity Diagram, MDA approach, Semantic Composition, Workflow, Grid Services.

1 Introduction

1.1 Motivation

The grid [1] is a technology that facilitates dispersed computation models based on resources situated in different administrative domains. Combined with the application of the open Grid Service Architecture [1], the resources of the Grid will be accessible as Grid services providing well defined functionalities. A Grid service is usually available through some uniform protocols and interfaces[1]. These services constitute a powerful basis for modern and scientific applications development. In order to enable users to compose their applications without taking care of the lower level details, the concept of Grid workflow has emerged as a method for modeling complex and scientific application [2]. The problem of building such applications requires finding and orchestrating appropriate services that is frequently a non trivial task for a developer. This is due to the very large number of available services and the different possibilities for constructing a workflow from matching services. Therefore, we propose in this paper an approach for automatic composition of workflow applications of grid services using UML activity diagrams [3]. Recently, several solutions have been proposed to compose applications from Grid services such as works presented in [2,4]. However, the proposed solutions need interaction with user and guidelines or rules in the design of the composed applications. In consequence, the resulting source code is neither re-usable nor it promotes dynamic adaptation facilities as it should. For applications composed of Grid services, we need an abstract view not only of the offered services but also of

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 615–626, 2009.
© Springer-Verlag Berlin Heidelberg 2009

the resulting application. This abstraction allows in one hand the reuse of the elaborated application and on the other hand reduces the complexity of the composed applications. There are several architectural approaches for distributed computing applications [5] which make easy the development process. However, these approaches need rigorous development methods to promote the reuse of components in future Grid development applications. It has been proven from past experiences that using structured engineering methods makes easy the development process of any computing system and reduce the complexity when building large Grid applications. To reduce this complexity and allow the reuse of Grid service applications, we adopt a MDA approach. The MDA [5] approach developing starts with defining high-level models in UML [3], defines conversion rules from UML to target platform, and then use code generation to derive much of the implementation code for desired platform.

1.2 Our Contribution

We propose a new approach to built Grid services applications by following OMG(s) principals of the MDA in the development process [6]. In this approach, we are interested to compose and model workflows from existing Grid services. The workflow modeling identifies the control and data flows from one depicted Grid service's operation to the next to built and compose the whole application. To model and express the composed workflow of Grid services, we define and present in this paper an UML profile based on Domain Specific Language (DSL) for customizing UML activity diagrams for the systematic composition of workflows from Grid services. In addition, we propose a set of UML activity diagram patterns to express the different forms of matching services in a workflow. The provided model forms the Platform Independent Model (PIM) of the proposed MDA approach.

Fig. 1 presents the architectural view of the proposed approach [6]. At the first step of the approach, the user specifies its problem, i.e. the result that it wishes to get from

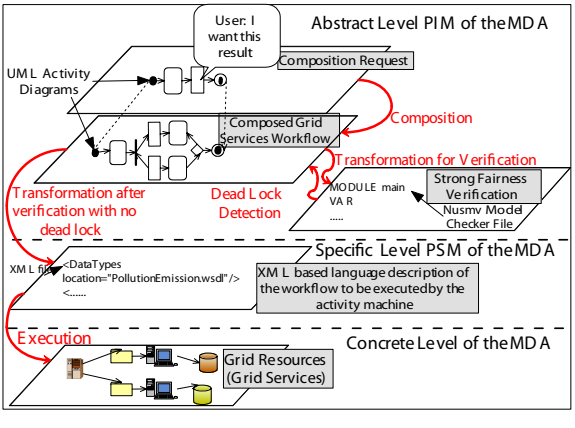


Fig. 1. Architectural view of the approach

the workflow composition, by modeling a composition request using UML activity diagrams. This request will be refined by the composition system to built the composed workflow from available Grid services. Before being executed, the resulted workflow will be be transformed to NuSMV model checker [7] file to verify the workflow reliability against the strong fairness property. As a next step, the resulting Grid services workflow model will be transformed into a XML based description language of workflows to be executed by a dedicated workflow engine using the Grid resources.

1.3 Paper Reminder

This paper is organized as follows. Section 2 presents the related works. Section 3 details the proposed UML profile for systematic Grid Services composition and section 4 presents the different workflow patterns for matching Grid services. Section 5 illustrates the composition process based on the proposed UML profile and workflow patterns. Finally, section 6 concludes the paper and proposes areas for further research.

2 Related Works

Several works and researches were carried out in the field of Systematic composition of workflows of Web and Grid services like works presented in [2,8,4]. The authors in [8] have proposed a Model Driven Approach for composing manually Web services. They were based on UML activity diagrams to describe the composite Web service and on UML class diagrams to describe each available Web Service. The user depicts the suitable Web service and match it in the workflow representing the composite Web service using UML activity diagrams. This approach would have been better if the composition were automatically elaborated, since the number of available services is in increase with the existence of several forms and manners to compose such services. Based on domain ontology description and a specific domain language for UML activity diagrams, we propose, in our approach, a systematic composition of workflow applications from Grid services rather than a simple service composition. In the field of Grid services composition the most similar and related work is the work presented by Gubala and Bubak in [2,4]. In this work, the authors have developed a tool for semi automatic and assisted composition of scientific Grid application workflows. The tool uses domain specific knowledge and employ several levels of workflow abstractness in order to provide a comprehensive representation of the workflow for the user and to lead him in the process of possible solution construction, dynamic refinement and execution. The originality of our contribution, relatively to this work, is that first we save the user from the dynamic refinement and execution as we propose a MDA approach which separates the specific model from the independent model. Second we use UML activity diagrams to deliver the functionality in a more natural way for the human user as they provide an effective visual notation and facilitate the analysis of workflows composition from Grid services. The use of UML activity diagrams in the description of workflow application is argued in several works such as works presented in [9,10,11].

3 UML Profile for Systematic Grid Services Composition

As UML is the core of the MDA [5] and their modeling elements provide an abstract description of systems; we use its activity diagram language [3] for modeling composed workflow from Grid services. However, the semantics of UML modeling elements are not well defined to be used in a specific domain. To overcome this deficiency, the UML specification defines the mechanisms for specializing semantics of modeling elements for a particular domain. In this section, we define an UML profile based on Domain Specific Language (DSL) for customizing UML activity diagrams for the systematic composition of workflows from Grid services. The objectives of the definition of this Domain Specific UML profile includes:

- The user is exposed only to domain specific UML modeling elements,
- The language concepts have domain specific interpretation,
- models may be enriched with information which are used by workflow engine to execute the abstract workflow model.

In our DSL and as it is illustrated in Fig. 2, an activity of an UML activity diagram represents a Grid service's operation, while object flows represent the types of results which flow from one activity to another. Effects binding two operations are represented with control flows. The name of an activity in the diagram represents the name of the Grid service's operation. This name must be specified as a Grid service could have more than one operation often called interface which are specified in its relative WSDL file [12]. There are two different types of activities: *yet-unresolved* activities and *established* activities of the composed workflow. The former represent the need for a Grid Service's operation to be inserted in order to complete the workflow. however, the latter represent abstract operations that are already included into the workflow. As there are two different activity types in a Grid service workflow model, an activity needs to be typed and specified. To fulfil this, we propose to use the DSL modeling element *invoke* to stereotype an established activity which is used to invoke an external Grid services's operation and *yet-unresolved* to stereotype activities which are not yet resolved.

Object nodes of an established activity are *data* stereotyped. Unknown input and output for a *yet-unresolved* activity are *unknown* stereotyped. In our UML profile, an

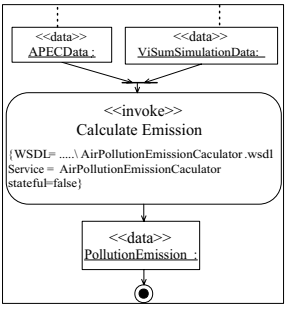


Fig. 2. Grid service's operation Pattern

object node could be related to a final node as composed workflow of Grid application should always deliver a result.

To produce a complete workflow model, able to invoke remotely the integrated Grid services instances, we need to specify additional information like service name, WSDL file [12] and service state. According to the UML modeling pattern [5], we register these information as tagged values. It should be noted, that the information provided by tagged values does not break the abstract level of workflow modeling. This is because they only indicate, for example, the fact that the same instance of a *stateful* Grid service should be used or they indicate the name of the Grid service to invoke, but they do not point to any *specific* instance.

4 Grid Services Workflows Composition Patterns

Based on UML concepts and extensions, we identify, in this section, how UML activity diagrams support some of basic Grid service composition patterns. These patterns are essential in the systematic building of workflow applications from Grid services. The use of these patterns depends on the number of the depicted Grid Service's operations and their inputs and outputs. These operations are results of the semantic research elaborated by the ontological Grid services registry which is responsible for storing and managing documents containing descriptions of syntax and semantics of services and their operations expressed in a OWL-S file [13]. This research is invoked by a request given by the composition system in order to complete an unresolved activity in the workflow. The Grid service registry provides zero, one or more operations producing the intended output. Operations are depicted to be inserted in the workflow interactively with the user. The composition process will be briefly presented through an illustration in section 5. As the composition is systematic, we are based on a set of predicates and functions which allow the selection of the right pattern to use in the workflow model. In the following we define formally these functions and predicates which depend on the number of selected Grid services, operations and their inputs.

4.1 Formalization

Let $\Omega_{activity}$ be the activities set and Ω_{object} be the objects set of a given activity diagram. The two sets are assumed to be finite.

– *Nb-Of-Input* function is used to denote the number of inputs of a given activity *Ac* belonging to $\Omega_{activities}$.

$$\begin{array}{c} \textit{Nb-of-Input:}\ \Omega_{activity} \rightarrow IN \\ \mathrm{Ac} \mapsto n \end{array}$$

– *Nb-source* is function which denotes the number of sources of a given object *o* belonging to Ω_{object}.

$$\begin{array}{c} \textit{Nb-source:}\ \Omega_{object} \rightarrow IN \\ \mathrm{o} \mapsto n \end{array}$$

– An activity may have one or more than one input and for each case, it requires a specific composition pattern. To capture this we define a predicate *require-one-data* presented as follows:

$$\textit{require-one-data}: \Omega_{activity} \rightarrow \{true, false\}$$
$$\textit{require-one-data}(\text{Ac}) = \begin{cases} \text{True if } \textit{Nb-of-Input}(\text{Ac}) = 1 \\ \text{false if } \textit{Nb-of-Input}(\text{Ac}) > 1 \end{cases}$$

– When the Grid registry provides more than one operation able to produce the required result, the composition requires a specific pattern. To capture this, we define a predicate *require-one-data* presented as follows:

$$\textit{require-one-source}: \Omega_{object} \rightarrow \{true, false\}$$
$$\textit{require-one-source}(\text{o}) = \begin{cases} \text{True if } source(o) = 1 \\ \text{false if } source(o) > 1 \end{cases}$$

– *succ(Ac)* is a recursive function that provides the set of all successor activities of *Ac* until the final node.

$$\textit{Succ}: \ \Omega_{activity} \rightarrow \Omega_{activity}$$
$$\text{Ac} \mapsto \text{set of successors of Ac}$$

$succ(Ac) = \{Ac' \in \Omega_{activity} / outgoing(Ac) \subseteq incoming(Ac') \text{ and } Ac' \neq finalnode\} \sqcup \{succ(Ac')\}$. Where:
incoming(Ac) is the set of the activity diagram edges entering in Ac and
outgoing(Ac) is the set of the activity diagram edges leaving Ac.

– As the workflow composition of Grid services is done in a down-top manner, the predicate *in-loop*(Ac) is essential to test if the inserted activity is implicated in a loop. The predicate is defined as follows:

$$\textit{in-loop}: \Omega_{activity} \rightarrow \{true, false\}$$
$$\textit{In-loop}(\text{Ac}) = \begin{cases} \text{True if } \exists Ac' \in succ(Ac) \text{ and } \exists \text{ an edge } e \in outgoing(Ac'), \\ e \in incoming(Ac) \\ \text{false else.} \end{cases}$$

– When composing workflows of Grid services, a specific matching based on semantic comparison could provide two or more different Grid services performing each of them the required operation. Let *service(Ac)* be a function providing the set of Service's names of the operation represented by the activity Ac. The service's names are depicted from the activity tagged value: *Service*. To capture the fact that a depicted operation may belong to different services, we define the predicate *alternative-service(Ac)* as follows:

$$\textit{Alternative-service(Ac)}: \Omega_{activity} \rightarrow \{true, false\}$$
$$\textit{Alternative-service}(\text{Ac}) = \begin{cases} \text{True if } \mid service(Ac) \mid > 1 \\ \text{false if } \mid service(Ac) \mid = 1. \end{cases}$$

4.2 Sequence Pattern

Description. If the Grid registry sends, to the composition system, one Grid service's operation that is able to produce the required result or the user selects one operation from the provided operation set. Let *Ac* be a single abstract operation (activity) which will be inserted in the workflow, with *require-one-data(Ac)= True*. This operation may also require some data for it self and thus it may introduce a new unresolved dependency. So, we use a follow-up method to built a simple pipeline-type sequential workflow: a *sequence pattern*.

Proposed Solution. We propose to use sequential activities which are related with control flow (non-data operations dependency) or object flow (data operation dependency) (See Fig. 3).

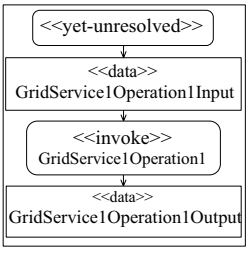


Fig. 3. Sequence Pattern

4.3 And-branches Pattern

Description. The and-branches is introduced when *require-one-data(Ac)= false*, with *Ac* is an abstract operation having more than one input. This pattern is based on the Synchronization pattern presented in [9].

Proposed Solution. We start with object nodes, representing alternative operation inputs, which flow to a join node. The latter is linked to the abstract grid service's operation. This operation introduces some unresolved dependencies in the workflow. Semantically, several services instances are invoked in parallel threads, and the join will wait for all flows to finish. As illustrated in Fig. 4, the operation of the Grid service ***GridService1Operation1*** needs two inputs data ***GridService1Operation1Input1*** and

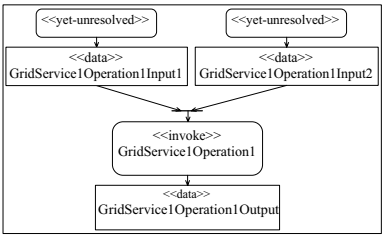


Fig. 4. And-branches Pattern

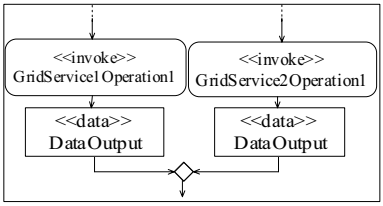


Fig. 5. Alternative branches Pattern

GridService1Operation1Input2. The relative pattern produces two parallel threads in the workflow.

4.4 Alternative Branches Pattern

Description. When *require-one-source(o)= false*, with *o* is an object data representing the required output. Thus, we introduce the alternative branches patterns. This pattern combines the Exclusive Choice and Simple Merge patterns presented in [9]

Proposed Solution. In this pattern, each alternative service's operation in liked to object node representing the required input which flow to a merge. Semantically, several services instances are invoked in parallel threads and the merge will only wait for the first flow to finish. In Fig. 5, we distinguish two different services's operations ***GridService1Operation1*** and ***GridService2Operation1*** providing the same output data ***DataOutput***.

4.5 Loop Pattern

Description. The loop construct is widely used for simulation based scientific computation. In the composition process, when *in-loop(Ac)=true* and *Ac* is the current activity to insert, we introduce the pattern.

Proposed Solution. We propose to use the loop node proposed by UML2.0 specification [3]. A loop node is represented with a rectangle having the loop tag in its high left corner as shown in the Fig. 6. A guard condition, testing the end of the loop, is put at the life flow of the diagram portion to iterate (*Ac* is the top activity of the flow and Ac' is the bottom one). The guard condition is provided by the Grid registry as the pre-condition of the operation. Fig. 6 shows a portion of the example of city traffic pollution analysis workflow that contains a loop. This loop is involved as the application iterates in order to analyze the possible traffic scenarios.

4.6 Alternative Services Pattern

Description. When the predicate *Alternative-service* is applied on the current activity *Ac* to be inserted in the workflow and it is evaluated as true, the activity is involved in the workflow using the pattern *alternative-service*.

Proposed Solution. In this solution the activity to insert is modeled as a composed super activity with a specified input data object and specified output data object (Fig. 7).

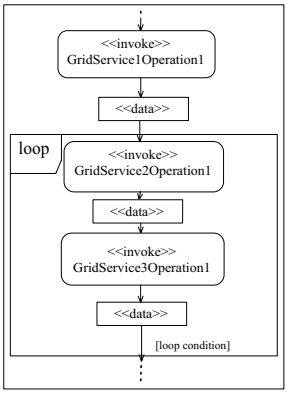


Fig. 6. Loop Pattern

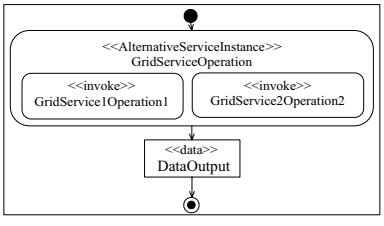


Fig. 7. Alternative services Pattern

The super activity is stereotyped as *AlternativeServiceInstance* to indicate that its task may be accomplished by a set of alternative service's instances. These alternative service instances are described with sub-activities. The sub-activities shall be grid service instances and thus stereotyped as *invoke*. It was up to decision mechanism of the workflow execution engine to choose which service instance in a such given workflow node is to be invoked and executed. In Fig. 7, the data ***DataOutput*** is provided from ***GridServiceOperation*** service operation which could be ***GridService1Operation1*** provider or ***GridService2Operation2*** Provider.

5 Illustration of the Composition of Workflows from Grid Services

In the following, we illustrate the composition process through the example of the domain of the city traffic pollution analysis. This application, as presented in [4], targets the computation of traffic air pollutant emission in an urban area.

We define the workflow composition problem as a non empty set of results that the user expects from the workflow execution.

– The user builds its composition request using our composition tool graphical interface by specifying its problem using the extended activity diagram notation describing the initial workflow.

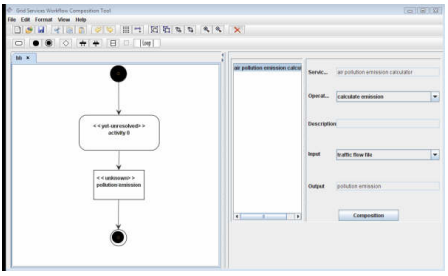


Fig. 8. Initial workflow as a composition request

Fig. 8 shows an example of initial workflow that represents a composition request for the results of the pollutant emission due to the city traffic. The desired result is described by the rectangle representing the object node (unknown result) in the relative activity diagram. For each *unknown* result, one dependency is defined as a workflow which depends on some Grid service's operation *yet unresolved*.

– The system analyzes the *unknown* result specified in the activity diagram and requests the Grid registry for a Grid service's operation having the result as output. If the Grid registry returns an operation, the composer introduces the operation in the workflow and continues to resolve the *unknown* other results. In the Grid registry, services are described in an ontological form with statements regarding the service operation's inputs, outputs, preconditions and effects (the IOPE set) [13]. Through these notions, the composer system is able to match different operations into a workflow following a reverse traversal approach. Thus, and by associating the required data with the produced output, the composer constructs a data flow between operation using workflow patterns and our UML profile. The composer may also use a specific notion of effect that may bind two operations together with non-data dependency.

Fig. 9 represents the workflow of the computation of traffic air pollution analysis after one step of composition. In this case, the operation *AirPollutionEmissionCalculator* needs more than one inputs, thus the composer uses the synchronization

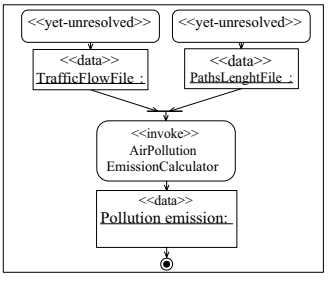


Fig. 9. An example of workflow after one step of composition

pattern. Other *yet-unresolved* activities are added in order to keep the workflow valid with regard to the activity diagram syntax rules.

– For every unresolved dependency i.e *unknown* data and *yet-unresolved* activity, the composer contacts the ontological registry in order to find the suitable service's operation and data that could be inserted in the workflow to fulfill the dependency. The composition process will stop if all dependencies are resolved or if the Grid registry fails to find suitable operations.

6 Conclusions

In this paper, we have detailed a set of composition patterns for integrating and matching, systematically, Grid services in a workflow. In addition, we have defined a set of functions and predicates which are used in the composition process to depict the right pattern to involve in the workflow. For the composition and the execution of composite workflows, we have proposed an MDA approach [6] based on UML activity diagrams [5]. We have, also, proposed an UML profile for customizing UML activity diagrams to compose, systematically, workflows from Grid services. This profile extends UML activity diagrams to meet some Grid services workflow modeling needs by proposing a Domain Specific language (DSL). The composition process was illustrated through the example of city traffic pollution analysis domain [4].

We have developed and implemented the composition tool and we are working actually on the implementation of the verification system using NuSMV model checker [7] and of the workflow execution system as well. The latter invokes and executes the depicted Grid service instances and manages the control and data flows in a run time environment relatively to our proposed activity diagram semantics.

References

1. Foster, I., Kesselman, C.: Grid services for distributed system integration. IEEE Computer (2004)
2. Bubak, M., Guballa, R., Malawski, M., Rycerz, K.: Workflow composer and service registry for grid applications. Future generation Computer Systems (2005)
3. Group, O.M.: Uml 2.0 superstructure specification. Technical report (2005)
4. Guballa, R., Hoheisel, A., First, F.: Highly dynamic workflow orchestration for scientific applications. CoreGRID Technical Report, Number TR-0101 (2007)
5. Group, O.M.: Model driven approach. Technical Report omrsc/2001-07-01 (2001)
6. BendalyHlaoui, Y., JemniBenayed, L.: Toward an uml-based composition of grid services workflows. In: AUPC 2008, 2nd international workshop on Agent-oriented software engineering challenges for Ubiquitous and Pervasive Computing, ACM Digital Library (2008)
7. Cimatti, A., Clarke, E., Giunchiglia, E., Giunchiglia, F., Pistore, M., Roveri, M., Sebastiani, R., Tacchella, A.: Nusmv version 2: An opensource tool for symbolic model checking. In: Brinksma, E., Larsen, K.G. (eds.) CAV 2002. LNCS, vol. 2404, p. 359. Springer, Heidelberg (2002)
8. Gronomo, R., Jaeger, M.: Model driven semantic web service composition. In: APSEC 2005 (2005)

9. Dumas, M., Hofsetde, A.: UML activity diagrams as a workflow specification language. In: Gogolla, M., Kobryn, C. (eds.) UML 2001. LNCS, vol. 2185, p. 76. Springer, Heidelberg (2001)
10. Eshuis, R., Wieringa, R.: Comparing petri net and activity diagram variants for workflow modelling: A quest for reactive petri nets. In: Ehrig, H., Reisig, W., Rozenberg, G., Weber, H. (eds.) Petri Net Technology for Communication-Based Systems. LNCS, vol. 2472. Springer, Heidelberg (2003)
11. Gardner, T.: Uml modelling of automated business processes with a mapping to bpel4ws. In: Cardelli, L. (ed.) ECOOP 2003. LNCS, vol. 2743. Springer, Heidelberg (2003)
12. W3C: Web services description language (wsdl). Technical report (2001)
13. Coalition, O.S.: Owl-s: Semantic markup for web services. Technical Report version 2.0 (2001)

A Case Study of Knowledge Management Usage in Agile Software Projects

Anderson Yanzer Cabral, Marcelo Blois Ribeiro, Ana Paula Lemke, Marcos Tadeu Silva, Mauricio Cristal, and Cristiano Franco

PPGCC, PUCRS, Av. Ipiranga 6681, Porto Alegre, Brazil
{anderson.cabral,marcelo.blois,ana.lemke,marcos.tadeu}@pucrs.br
{mauricio.cristal,franco.cristiano}@gmail.com

Abstract. Agile Methodologies promote a group of principles which differ from Traditional Methods. In this way, one concrete difference is the manner of how the knowledge is managed during a software development process. Most proposals to knowledge management have been generated for Traditional Methods but have failed in Agile Projects because they focus on explicit Knowledge Management. This paper aims to present a case study with a detailed contributions taken from Lessons Learned for some issues related to Knowledge Management in a distributed project that make use of Agile Methodologies.

Keywords: Agile Methodologies, Knowledge Management, Case Study.

1 Introduction

In the software engineering field, the issue of how software development should be organized in order to deliver faster, better, and cheaper solutions has been widely discussed in the last years. Many methodologies, practices, techniques and tools have been suggested for process improvement.

Recently, many suggestions have come from experienced professionals, who have labeled their methods as "agile software development". This movement has had a huge impact on how software is developed worldwide [9].

Agile methodologies promote a group of principles for software development which differ from traditional methods, also known as plan-driven or Tayloristic [5], [7]. Some authors, such as [2], consider agile methodologies as a reaction against the bureaucracy of traditional methods. Actually in the literature we can find a list of topics where there are differences and similarities between agile and traditional methodologies [1], [5], [9].

One of the differences is the way in which the knowledge is managed during software development processes. Traditional approaches make an intensive use of documents to capture and represent knowledge gained in the activities of a software project lifecycle, ensuring product and process conformance to prior plans, supporting quality improvement initiatives, and satisfying legal regulations [5]. On the other hand, agile approaches suggest that most of the written documentation can be

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 627–638, 2009.
© Springer-Verlag Berlin Heidelberg 2009

replaced by enhancing informal communication among team members and between the team and the customers with a stronger emphasis on tacit knowledge rather than explicit knowledge [6]. However, prioritizing communication in agile methodologies does not mean to disregard formal documentation.

There are some proposals to manage the knowledge generated during traditional software development process, such as [3], [11], [13], [18], [21]. However, most of these approaches are not suitable for agile methodologies, because they focus on explicit Knowledge Management, while agile methodologies make emphasis on tacit knowledge interchange.

This paper presents some issues related to Knowledge Management in projects that make use of agile methodologies. These issues were identified during a case study in agile projects. This case study will be detailed in subsequent sections from where contributions will be pulled out in the Lessons Learned style, which will address among other things, problems associated with difficulties in the use of agile methodologies in distributed environments; problems generated by the prioritization of the conversation as the main way of sharing knowledge; and problems in the project's documentation.

This work is structured as follows: in Section 2 the main concepts used in this paper are explained. Section 3 presents the case study with its instruments and data analysis. In Section 4 the discussions and lessons learned extracted from the case study are presented. The Section 5 shows the related work founded in the literature. The conclusions are discussed in Section 6.

2 Background

In a competitive world, it is primordial that organizations hold the capacity to accomplish changes to increase their advantages. Nevertheless, due to the lack of understanding about the company itself, the fulfillment of those changes not always bring the expected results. In other words, the achievement of the changes that target benefits to the organization is hampered by its scarcity of knowledge of both, the way the business processes are conducted and its own organizational structure.

In such case, increasing organizational knowledge raises an opportunity to organizations become more competitive [15]. The organizational knowledge includes, but is not limited to, knowledge related to business processes, knowledge about the relationship among the various organizational sectors, and knowledge about market, technologies, customers and competitors [8]. Such knowledge has been considered the organizational intellectual capital, it must be managed efficiently, ensuring its preservation and enabling its constantly evolution, which can be guaranteed through a policy of Knowledge Management.

Knowledge Management is a large interdisciplinary field and as a consequence, there is an ongoing debate as to what constitutes Knowledge Management. For this paper, we use a definition that is common sense. Davenport & Prusak [8] have defined Knowledge Management as ‘‘a method that simplifies the process of sharing, distributing, creating, capturing and understanding of a company’s knowledge”.

For software development organizations, the main asset is the intellectual capital and not buildings and machines [20]. One mean suggested to overcome this problem is the increased focus on Knowledge Management.

Knowledge Management in software development organizations is a large field with several disciplines that can influence their results. In [4] it is explained a systematic review about Knowledge Management in software engineering. This study discuss on the main implications for deciding which Knowledge Management methodology is adequate whether a company seeks to having agile development processes in place, or relies on traditional development methods such as the waterfall process.

When traditional methodologies are in use, the objective of the Knowledge Management is to try as much as possible to turn tacit knowledge into explicit knowledge. So, the tasks related to the management, like acquisition, storage, update, share, can be facilitated.

For agile methodologies, the focus is on working with tacit knowledge. It is different to avoid working with the explicit knowledge. The effort that would be spent for convert tacit knowledge is the main issue considered here.

3 Case Study

This research is exploratory and based on case study. The case study was developed in two distinct organizations - the *Owner Organization* and the *Offshore Organization*. We analyzed the work of the teams that worked in two different but correlated projects in the cited organizations. The first project (*Project 1*) consisted in the development of a new software and the other one (*Project 2*) was a project for software maintenance (both applications aim to solve the same business opportunity).

In the next sections, we provide a detailed discussion about the context of each organization (Section 3.1), the research instruments used for data collection (Section 3.2), and the considerations obtained from data analysis (Section 3.3). As it can be seen in Section 3.2, this research has taken advantage of diverse qualitative and quantitative data collecting techniques, such as: questionnaires, document analysis, and script for semi-structured interviews.

3.1 Context

In this context, an organization A has developed a Product P which consists in a software which controls its importations. Thus, it is possible to think about the better way of decrease these taxes. Furthermore, this Product is divided in two Sub-Products: Sub-Product 1, which implements the rules and the process for tax reduction; and Sub-Product 2, which shall expedite the process of clearance of these imported goods, reducing the rigidity of surveillance.

These Sub-Products have been implemented through outdated technologies, and do not automate some new business rules, entailing in a great amount of manual work. Anyway, these Sub-Products development has been maintained continuously by Project 2.

However, due to the needs of the organization, a new project (Project 1) with up-to-date technologies aims to provide an automation of all business rules, getting connected with the interfaces of communication systems of Government. Because the large number of stakeholders, the several system's interface, and some accounts access restrictions, this new project has a high level of complexity, showing the need for development between two distributed teams.

The first team, called PUC-Team, is located at University of Rio Grande do Sul. The other one can be found placed at the Federal University of Pernambuco and stands for UFPE-Team. Both teams have experienced collaborators with the SCRUM Master Certify. However, apparently there were problems in establishing a channel of communication between the teams, once there have been several weekly virtual and phone conferences between them. Both projects use Agile Methodologies, but actually, the Project 1 has been planned to use a proprietary methodology, later, it has changed to use Agile Methodologies, passing through a moment of transition between these two methodologies.

The documentation of Project 1 constitutes of some artifacts made during its design phase. It has happened because the use of the proprietary methodology at the beginning of the project and the proprietary methodology has been based on traditional software engineering approaches. The entire architecture of the Project 1 have been done and evaluated by a specialized software architecture validation team, located far from the two other teams.

3.2 Research Instruments

A systematic approach is required for developing measuring instruments. To evaluate the profile of the people involved with the projects, as well as to identify the responsibilities of each one during the projects execution, we defined a survey named "team overview survey". The team overview survey was divided into three parts: (i) general questions; (ii) questions about team members' experience on different software development methodologies; and (iii) questions about specific features of the projects.

The questions included in the first part intended to verify: the team members' education level, how many years each member had of professional experience in Information Technology; the current working relationship of the team members with the organizations (if employee, contractor or trainee/intern); how many months each one have been working for the organizations; the team members' English language expertise; and the team members' age.

The second part of the survey aimed to evaluate the team members' expertise in agile, hybrid and traditional methodologies. For this purpose, each team member was asked about: the years of professional experience working with each methodology; the number of distinct projects working with the methodologies during the years of professional experience; and the usability of the documentation generated during the projects (if the documentation generated was always useful, useful most of the time, useful sometimes or it was not useful in the project tasks). Each team member answered only the set of questions related to the methodologies he/she had previous experience.

Team members with experience in agile methodologies answered four additional questions. The additional questions intended to check: a) which agile methodology was used on previous projects; (b) if the company provided some formal training in agile development processes for the team; if the team members had some certification on agile methodologies; and what characteristics of agile methodologies were present in the processes used in previous projects (according to team member perception).

The main characteristics of agile processes were defined based on agile methodologies literature [7]. They are: active user involvement, close collaboration of business people and developers, managing requirements throughout the development, testing integrated throughout the lifecycle. For each characteristic, the team members needed to select "yes" (indicating that the characteristic was present in the projects) or "no" (indicating that the characteristic was not present in the projects). The last question included into the second part of the survey aimed to identify the team members' preferences among the methodologies (which methodology the team members would select to use in a new project).

The third part of the survey intended to verify the roles and responsibilities of the team members into the projects. The questions included in this part aimed to identify: (1) the team members' roles; (2) the team member's domain expertise; (3) the estimated allocation of each team member at the projects; (4) the perception of each team member about the process used in the projects (if the defined process was mainly a traditional or an agile methodology); (5) and the perception of the team members about the usability of the documentation generated during the projects.

The survey was answered by 15 of 17 team members who have participated in *Project 1* in the *Owner Organization* and the *Offshore Organization* and by 4 of 6 members who have participated in the *Project 2* in the *Owner Organization*. Since all employees allocated in the *Project 2* were also allocated in the *Project 1* (with partial allocation in each one), 15 questionnaires were answered. The results analysis is described in the next section.

In addition to the questionnaire, we interviewed some team members for collecting data. The main goal of made interviews was to discuss about the documentation generated and used during the projects. For this purpose, the employees were asked about the kind of documentation generated in the projects, who created and make available the documentation, and how tacit knowledge was exchanged among them (the communication channels used to exchange tacit knowledge). The interviews were done in a semi-structured way. Fifteen employees were selected to be interviewed. They were interviewed in individual meetings of approximately one hour. All interviews were made in Portuguese language. The analysis of the interviews can be seen in the next section.

3.3 Data Analysis

The purpose of this section is to analyze the data obtained through the questionnaire and interviews described in section 3.2. We will present just the relevant information obtained through a statistical analysis. The aim was to identify which trends and correlations between the data analyzed could bring some contribution within the purpose of this study, which has as scope: Knowledge Management projects in agile software development.

The focus of this data analysis is to understand relevant aspects making a comparison between agile and traditional methodologies, focusing on the change of use of explicit knowledge for a greater use of tacit knowledge.

In this sense, issues related to documentation, conversation, knowledge representation form, and how knowledge is acquired and shared, become the core of the analysis.

In the figures below some relevant of data obtained through data analysis of questionnaires and interviews.

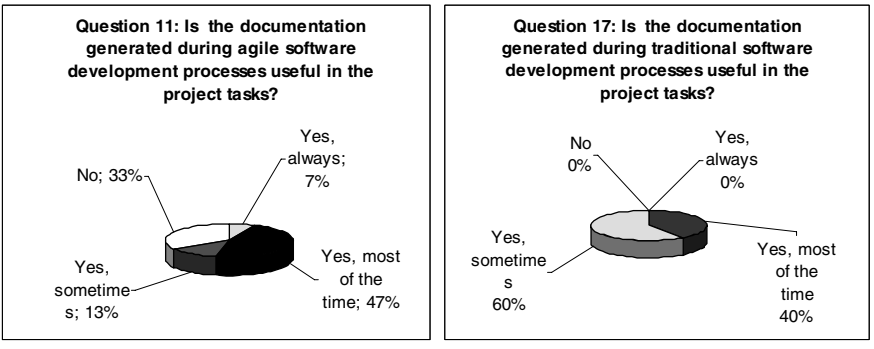


Fig. 1. Questions 11 and 17 of the questionnaire

Figure 1 illustrates the responses to question 11 and 17 of the questionnaire. Question 11 tried to check with members of the team as they consider useful the documentation in an agile process. This is important because it can show the confidence degree that the team has about the documents generated during the software development process. There were more responses indicating that they do not consider useful (33%) than always useful (7%).

The question 17 of the questionnaire tried to check with members of the team as they consider useful the documentation in a traditional process. There were no responses indicating that they do not consider useful (0%) nor that is always useful (0%).

Figure 2 illustrates the responses to question 5 and 7 of the interview. The Question 5 examined whether the respondents consider updated the documentation generated during the process of software development. Almost all (93%) consider the documentation outdated.

The Question 7 examined the priority which the respondents use to obtain knowledge about the project, if the documentation or colleagues and stakeholders first. The majority (73%) consulting the documentation first. It should be emphasized that this issue defines the order of priority preferred by the interviewee in the moment for to get some knowledge.

Figure 1 shows the results on the usefulness of the documentation in the software development process. What draws attention in these data is that the respondents considered the most important the documentation in agile methodology. During the interviews we realized that these responses are linked to the fact that in traditional methodologies they do not consider much of the documentation useful. However, the documentation that is used in agile projects is considered more useful.

The Pearson correlation coefficient found between issues 11 and 17 is 0.536. This value does not indicate a strong correlation between the two issues.

Figure 2 shows that the majority of interviewees considered the documentation outdated, but still uses the documentation before consulting colleagues. The Pearson

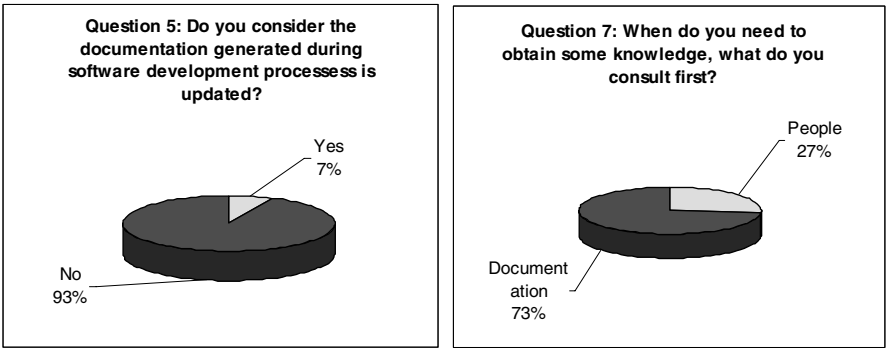


Fig. 2. Questions 5 and 7 of the interview

correlation coefficient found between issues 05 and 07 is 0.99. This value indicates a strong correlation between the two issues.

There were no significant correlations and relevant through quantitative analysis of the data analyzed. The main analyses that are present in the lessons learned were obtained through qualitative analysis of data from the questionnaire and especially from the interviews.

4 Discussions and Lessons Learned

According to the observations drawn from our case study, we suggest some items to be discussed as a result of findings and the need for further study and reflection on the following topics.

The use of agile methodologies in distributed software development intensifies the communication problems: in this case study the development process is characterized as Distributed Software Development (DSD), where the teams are situated in two sites and it has been perceived through interviews and monitoring of meetings (daily Scrum) some communication problems between the distributed groups.

The communication in DSD is, naturally, identified as a major difficulty [16]. In the context of projects agile communication becomes even more relevant because in this type of project the communication based on discussions is prioritized in relation to other forms of knowledge exchange. Korkala & Abrahamson [12] stand out that agile software development involves highly volatile requirements which are managed through efficient verbal communication.

As the communication is an important factor to be observed, in our case study was found that each group has its own repository of documents and artifacts that do not work in sync. There is no established practice for access and sharing of artifacts between repositories. That led members of the team to intensify communication among members of the two sites.

If there are mechanisms for sharing knowledge (repositories) that work so planned and coordinated between the groups, the need for communication through talks tends

to decrease. This will lessen the communication factor as a problem in this type of project, agile in DSD.

It is important to emphasize that we are not suggesting a greater use of the repositories and artifacts, conflicting with some agile assumptions, but what is available for groups should be properly shared, reducing the need for talks about information that is already available in one of sites.

Some communication problems between the two sites may occur because of cultural aspects. One site is located in the northern region of Brazil and another in the southern region. Cultural differences intensify conflicts in communication. At the company where the case study was conducted there are projects where a site is located in Brazil and another in India and communication is made in English. Imagine the communication problems in a project being implemented according to this format of agile methodology.

The intense use of informal communication on agile projects creates problems of knowledge acquisition and storage: Another aspect realized about the communication between team members and customers is that most of the talks held during the project are informal. This format makes the conversations more agile, but in compensation the knowledge generated during these talks remains only in the tacit way when, even in agile projects, many details must be recorded and stored for further consultations or confirmations [19].

An example of this perceived situation in the case study was in relation to requirements that were raised along to customers. Many interviewees (53%) reported that it is important to record the talks with customers, according to the constant changes and updates in the requirements. Often the talks are not recorded creating a conflict of information between the parties involved in the process.

This question takes up one of the main points discussed in this work, which is the need for a new perspective to a Knowledge Management methodology to be employed in agile development processes. This methodology should be centered on the tacit knowledge that circulates in the projects.

Because the goal is not to turn the communication process bureaucratic among stakeholders, a suggestion is to use mechanisms for automatic speech recognition to organize and summarize the talks. In other words, there would be a minimal effort by the team with almost no interference in how the talks are accomplished, but that would result in capture and records of the knowledge into a format that if it is used in conjunction with indexing mechanism and ranking would facilitate the consultations and the sharing.

Despite the documentation be reduced and outdated the team uses as a source of knowledge to extract the context of the area and reduce the direct communication: during the process of interviews it was noticed that the interviewees made some criticisms on the documentation available for being inadequate and often outdated.

At first, we could imagine that the respondents did not use the documentation to consider it outdated and inadequate. However, despite this situation the respondents use the documentation. They explain that at least the documents provide the context of the knowledge they seek. They also used to reduce the time for direct communication. That is, when they need to obtain some knowledge and if they have already consulted the documents, they believe that will reduce the time in discussions with colleagues and customers.

Most respondents commented that the documents at the beginning of the project are updated, but over the course of the project will become outdated. The requirement documents are the most cited among those who are outdated during the project.

Many respondents commented about the difficulty to contact with the customer. That is, when they needed the documents of requirements were outdated and then they needed to contact the customer that was not always available.

The way the requirements were treated in this project doesn't fit properly neither with the agile methodologies principles, nor of the traditional methodologies. If the project follows the agile principles the customer should be more accessible and available for the whole team. If he follows principles of traditional methodologies, thus the requirement documents should be actualized for the whole project.

Under the focus of Knowledge Management, when you adopt an agile methodology the choice is by increasing communication between the team itself and between the team and the client.

When you take a traditional approach the choice is by prioritizing the knowledge in explicit form, thus giving more emphasis to the use of artifacts and documents, which must be updated throughout the life cycle of the project [5].

Thus, when using a hybrid approach must be taken great care to avoid those items do not meet satisfactorily neither agile nor traditional methodology, as was the case of the requirements in this project.

There is a lack of definition of what is a hybrid methodology. Qumer & Hederson-Sellers [17] present a framework to try to determine the agility degree of some methodologies. The border between not be agile and be hybrid, and not be traditional and be hybrid is very tenuous and deserves a better debate in the community.

In this project as a matter of contract, and choice of company, the phase of requirements followed a traditional approach and the other phases included agile practices, featuring a development process that followed a hybrid approach.

5 Related Work

In this section we will examine some published works that address Knowledge Management in software development. Most of works do not refer their use as agile or traditional methodologies in development teams.

Even though it, is clear that most of the initiatives reported are for teams that use traditional methodologies.

Initiatives, such as [3], propose the use of an Experience Factory. It is based on the fact that software development projects can improve their performance (cost, schedule and quality) through the use of previous experiences.

In DaimlerChysler [21] a model of Experience Factory was developed for the software development sector. Experience Factory is a way to make the acquisition and sharing of knowledge in the form of experience. The difficulty of packing knowledge in the form of experience was identified as one of the main challenges for the project's success.

Other experiences such as the Infosys [18], a company located in India, which has about 1,000 software projects running at the same time and with about 10,200 employees, has created a methodology to share knowledge through paper's repositories

written by employees who are rewarded by the level of advertising they reach. The company also makes a directory location of experts available.

The Goddard Space Flight Center of NASA has also a number of initiatives, which comprise a broad strategy for managing knowledge; many of them stand out for capturing and sharing knowledge in the tacit way [13].

These initiatives cited above are reported in more "traditional" software development environments. However, the work developed by researchers from the Department of Computer Science at the University of Calgary [5], [14] focuses the Knowledge Management on agile methodologies for software development, which widely raised the issue about prioritization of the tacit knowledge use rather than explicit knowledge.

In Korkala & Abrahamsson [12] and Holz & Maurer [10] the agile aspect is addressed together with the issue of distributed development. That is, how to work the issue of Knowledge Management in distributed teams, because it has been handled mainly in its tacit form.

In Bjornson & Dingsoyr [4] a systematic review about Knowledge Management in software engineering was accomplished, where seven hundred and sixty-two articles were identified, of which 68 were studies in an industry context. Of these, 29 were empirical studies and 39 reports of lessons learned. One of the research's main implications is the distinction between two types of development which has implications in the strategy of Knowledge Management, namely traditional and agile development.

The works mentioned address initiatives and strategic aspects of Knowledge Management in software development. They evidence the distinction that must be made between traditional and agile project, but none of them goes deepen with case studies, in the Knowledge Management in agile projects, for instance.

6 Conclusions

This paper has discussed the Knowledge Management in agile methodologies on the basis of a case study. This case study showed some peculiarities, such as having two teams distributed, which adds a relevant factor to the analysis. Another particularity of the case study assumes that the company is using a hybrid approach.

The case study through questionnaires, interviews and meeting monitoring of the team work enabled an analysis that was presented in discussions format and lessons learned. The lessons learned section is the main contribution of this paper, presenting aspects that were discovered and are discussed from the case study.

The quantitative data analysis obtained in the case study was not as significant as the qualitative analysis of the obtained information.

In this paper and in related work we realize that there is a gap for Knowledge Management in agile methodologies. Manage the knowledge in traditional projects focused on the use of explicit knowledge is not an easy task. But in agile project management is even more complex, because managing the knowledge in the tacit form involves many subjective aspects.

As future work, the intention is to develop and propose for the same team a Knowledge Management methodology focused on mechanisms (semi) automatic to

acquisition and sharing knowledge. It is also intended to measure the performance of the team using this methodology in comparison how the team currently works, as it pertains to managing knowledge.

Acknowledgements. Study developed by the Research Group in Intelligent Systems Engineering Group of the PDTI, financed by Dell Computers of Brazil Ltd. with resources of Law 8.248/91.

References

1. Abrahamsson, P., Salo, O., Ronkainen, J., Warsta, J.: Agile software development methods – review and analysis. Technical Report 478. VTT Publications (2002)
2. Agerfalk, P.J., Fitzgerald, B.: Introduction. Commun. ACM 49(10), 26–34 (2006)
3. Basili, V., McGarry, F.: The experience factory: How to build and run one (tutorial). In: Proceedings of the 19th international Conference on Software Engineering, Boston, Massachusetts (1997)
4. Bjornson, F.O., Dingsoyr, T.: Knowledge management in software engineering: A systematic review of studied concepts, findings and research methods used. Inf. Softw. Technol. 50(11), 1055–1068 (2008)
5. Chau, T., Maurer, F., Melnik, G.: Knowledge sharing: Agile methods vs. tayloristic methods. In: Proceedings of the 12th IEEE International Workshops on Enabling Technologies (WETICE 2003), Infrastructure for Collaborative Enterprises. IEEE Computer Society, Washington (2003)
6. Cockburn, A., Highsmith, J.: Agile software development: The people factor. Computer 34(11), 131–133 (2001)
7. Cohen, D., Lindvall, M., Costa, P.: An introduction to agile methods. Advances in Computers, Advances in Software Engineering 62(66), 2–67 (2004)
8. Davenport, T.H., Prusak, L.: Working knowledge: how organizations manage what they know. Harvard Business School, Boston (1998)
9. Dybaa, T., Dingsoyr, T.: Empirical studies of agile software development: A systematic review. Inf. Softw. Technol. 50(9-10), 833–859 (2008)
10. Holz, H., Maurer, F.: Knowledge management support for distributed agile software processes. In: Henninger, S., Maurer, F. (eds.) LSO 2003. LNCS, vol. 2640, pp. 60–80. Springer, Heidelberg (2003)
11. Komi-Sirviö, S., Mäntyniemi, A., Seppänen, V.: Toward a practical solution for capturing knowledge for software projects. IEEE Softw. 19(3), 60–62 (2002)
12. Korkala, M., Abrahamsson, P.: Communication in distributed agile development: A case study. In: EUROMICRO 2007: Proceedings of the 33rd EUROMICRO Conference on Software Engineering and Advanced Applications, pp. 203–210. IEEE Computer Society, Washington (2007)
13. Liebowitz, J.: A look at nasa goddard space flight center's knowledge management initiatives. IEEE Softw. 19(3), 40–42 (2002)
14. Melnik, G., Maurer, F.: Direct verbal communication as a catalyst of agile knowledge sharing. In: ADC 2004: Proceedings of the Agile Development Conference, pp. 21–31. IEEE Computer Society, Washington (2004)
15. Nonaka, I., Takeuchi, H.: The Knowledge – Creating Company: How Japanese Companies Create the Dynamics of Innovation. Oxford University Press, Oxford (1995)

16. Pikkarainen, M., Haikara, J., Salo, O., Abrahamsson, P., Still, J.: The impact of agile practices on communication in software development. Empirical Softw. Engg. 13(3), 303–337 (2008)
17. Qumer, A., Henderson-Sellers, B.: An evaluation of the degree of agility in six agile methods and its applicability for method engineering. Inf. Softw. Technol. 50(4), 280–295 (2008)
18. Ramasubramanian, S., Jagadeesan, G.: Knowledge management at infosys. IEEE Softw. 19(3), 53–55 (2002)
19. Ruping, A.: Agile Documentation: A Pattern Guide to Producing Lightweight Documents for Software Projects. John Wiley & Sons, Inc., New York (2003)
20. Rus, I., Lindvall, M.: Guest editors' introduction: Knowledge management in software engineering. IEEE Softw. 19(3), 26–38 (2002)
21. Schneider, K., von Hunnius, J., Basili, V.: Experience in implementing a learning software organization. IEEE Softw. 19(3), 46–49 (2002)

A Hierarchical Product-Property Model to Support Product Classification and Manage Structural and Planning Data

Diego M. Giménez[1], Gabriela P. Henning[1], and Horacio P. Leone[2]

[1] INTEC (UNL-CONICET), Güemes 3450, Santa Fe S3000GLN, Argentina
dgimenez@fiq.unl.edu.ar, ghenning@intec.unl.edu.ar
[2] INGAR (UTN-CONICET), Avellaneda 3657, Santa Fe S3002GJC, Argentina
hleone@conicet-santafe.gov.ar

Abstract. Mass customization is one of the main challenges that managers face since it results in a proliferation of product data within the various organizational areas of an enterprise and across different enterprises. Effective solutions to this problem have resorted to generic bills of materials and to the grouping of *product variants* into *product families*, thus improving data management and sharing. However, issues like product family identification and formation, as well as data aggregation have not been dealt with by this type of approach. This contribution addresses these challenges and proposes a hierarchical data model based on the concepts of *variant*, *variant set* and *family*. It allows managing huge amounts of structural and non-structural information in a systematic way, with minimum replication. Besides, it proposes an unambiguous criterion, based on the *properties* of *variants*, for identifying *families* and *variant sets*. Finally, the approach can explicitly handle aggregated data which is intrinsic to generic concepts like *families* and *variant sets*. A case study is analyzed to illustrate the representation capabilities of this approach.

Keywords: Product Data Model, Multiple Levels of Abstraction, Product Properties, Product Classification.

1 Introduction

Today's enterprises are forced to offer products which fulfill individual customer needs. Because of the growing mass customization, industrial environments are characterized by a large product variety. Therefore, an efficient treatment of product data is required in order to deal with the huge amount of product-related information that is managed daily within an enterprise and exchanged across different enterprises.

Several authors have proposed alternative solutions to the management of massive product data, based on the concepts of Product Family and Generic Bill of Materials [1,2]. In this direction, Giménez et al. [3] presented a novel product ontology named PRONTO. The core of this approach is a three-level *abstraction hierarchy*, where the lower level concerns physical products, which are handled through the concept of *variant*. The upper level abstracts a population of *variants* with some similar

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 639–650, 2009.
© Springer-Verlag Berlin Heidelberg 2009

characteristics and is managed through the concept of *family*. The intermediate level, handled through the concept of *variant set*, represents a set of *variants* with many similar characteristics. Thus, the *abstraction hierarchy* is constructed by grouping *variants* into *variant sets* and them into *families* according to their "similarity". This abstraction approach is helpful to share and not replicating common information. It is also very useful when carrying out planning activities at different levels (strategic, tactical and operational) since aggregated information can be generated along the hierarchy. Nevertheless, some issues, like product classification have not been treated in this line of research.

In this paper an unambiguous criterion for product classification is formalized. Despite many works in the specialized literature addressed issues regarding classification [4], the novelty of this work lies in that the proposed classification mechanism is compatible with the approaches for managing common structural and non-structural information as well as generating product data of different granularity. Specifically, the classification criterion focuses on the *properties* that are defined in order to describe the attributes of *variants*.

The paper is organized as follows. The concepts, relations and conditions on which the proposal is based are explained in Section 2. Section 3 presents a case study to illustrate the representation capabilities of the proposed approach. To conclude, some final remarks are presented in Section 4.

2 Proposed Approach

2.1 Product Classification

The proposed representation relies on an *abstraction hierarchy* having three levels, which are represented in the class diagram of Fig. 1 by the **Variant**, **VariantSet** and **Family** classes.

According to the classification criterion to be formalized in this section, a *family* is defined as a set of actual products or *variants* with several common *properties*, having values within a specified range. Similarly, a *variant set* is defined as a subset of *variants*, within a given *family*, which have the same *properties* and whose values

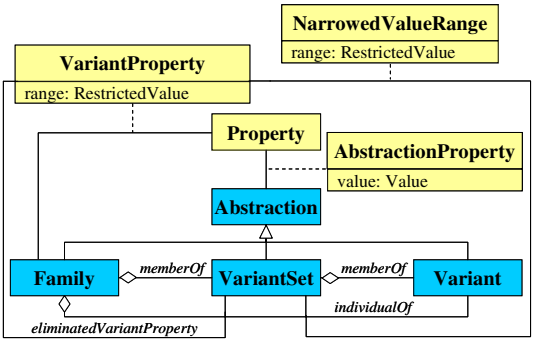


Fig. 1. The three-level abstraction hierarchy

are within a given range, included in the one defined for the *family*. Thus, the *variant set* notion can be seen as a subfamily concept.

Regarding the *abstraction hierarchy*, any concept (**Abstraction**) is of just one type (*family*, *variant set* or *variant*). If $\boldsymbol{A}$ denotes the entire set of *abstractions* and $\boldsymbol{I}$, $\boldsymbol{J}$, and $\boldsymbol{K}$ a partition of $\boldsymbol{A}$ representing the subsets of *families*, *variant sets* and *variants*, respectively; then, conditions specified in (1) must be satisfied.

$$(a)\ \mathrm{A} = \mathrm{I} \cup \mathrm{J} \cup \mathrm{K}\ ;\ (b)\ \mathrm{I} \cap \mathrm{J} = \{\ \}\ ;\ (c)\ \mathrm{I} \cap \mathrm{K} = \{\ \}\ ;\ (d)\ \mathrm{J} \cap \mathrm{K} = \{\ \} \tag{1}$$

In turn, each product instance or *variant* is a member of only one *variant set* and each *variant set* is a member of just one *family*. If $\boldsymbol{K}_j$ and $\boldsymbol{J}_i$ denote the set of members (*variants*) of *variant set j* and the set of members (*variant sets*) of *family i*, respectively, conditions prescribed in (2) and (3) are imposed.

$$(a)\ \forall k \in \mathrm{K}\ \exists\ j \in \mathrm{J} : k \in \mathrm{K}_j\ ;\ (b)\ \forall k \in \mathrm{K}, k \in \mathrm{K}_j \wedge k \in \mathrm{K}_{j'} \Leftrightarrow j = j' \tag{2}$$

$$(a)\ \forall\ j \in \mathrm{J}\ \exists i \in \mathrm{I} :\ j \in \mathrm{J}_i\ ;\ (b)\ \forall\ j \in \mathrm{J},\ j \in \mathrm{J}_i \wedge j \in \mathrm{J}_{i'} \Leftrightarrow i = i' \tag{3}$$

Consequently, each *variant* belongs to only one *family*, as clauses in (4) prescribe.

$$(a) \forall k \in \mathrm{K} : k \in \mathrm{K}_j \wedge\ j \in \mathrm{J}_i \Rightarrow k \in \mathrm{K}_i ; (b) \forall k \in \mathrm{K}, k \in \mathrm{K}_i \wedge k \in \mathrm{K}_{i'} \Leftrightarrow i = i' \tag{4}$$

where $\boldsymbol{K}_i$ denotes the set of individuals of *family i*.

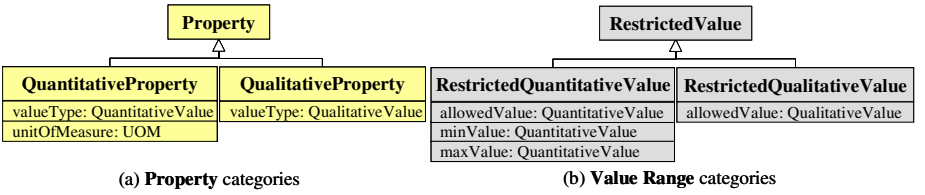
Properties play an essential role in product classification mechanisms. Two kinds of *properties* are proposed: (i) The ones associated with the individuals of a *family*, which are represented by the **Variant Property** concept (Fig. 1). This notion allows specifying for each *family* the *properties* of its population, as well as their ranges of possible values. Likewise, at the *variant set* level, the subset of *properties* shared by all its members is specified by removing those *variant properties,* associated with the corresponding *family,* that do not belong to the members of the *variant set* (*eliminated VariantProperty*). Though *variant properties* are specified at the *family* level, they generally assume values at the level of specific instances or *variants*. (ii) Properties that are particular or intrinsic to a generic concept, like *family* or *variant set*; hence, they are assigned values at the level of their definition. This notion is modeled by the **AbstractionProperty** association (Fig. 1). Property concepts are formalized in (5),

$$(a)\ \forall\ j \in \mathrm{J} : j \in \mathrm{J}_\mathrm{i} \Rightarrow \mathrm{P}_\mathrm{j}^\mathrm{K} \subseteq \mathrm{P}_\mathrm{i}^\mathrm{K}\ ;\ (b)\ \forall k \in \mathrm{K} : k \in \mathrm{K}_\mathrm{j} \Rightarrow \mathrm{P}_\mathrm{k} = \mathrm{P}_\mathrm{j}^\mathrm{K} \tag{5}$$

where $\boldsymbol{P}_i^K$ is the set of *variant properties* associated with *family i*, $\boldsymbol{P}_j^K$ the subset of *variant properties* shared by the members of *variant set j*, and $\boldsymbol{P}_k$ the set of *properties* of *variant k*. Moreover, the value range of a given *variant property,* specified at the *family* level, can be reduced when defining a *variant set* (**NarrowedValueRange**).

Properties are classified into *qualitative* and *quantitative* ones. In both cases the value type is specified, and for *quantitative properties*, the unit of measure must be indicated (see Fig. 2a).

The value range of *variant properties* is represented by the class **RestrictedValue** which, in turn, is specialized into quantitative and qualitative categories (see Fig. 2b). Within these subclasses a discrete set of allowed values can be specified. It is also possible to define a continuous range for quantitative values.



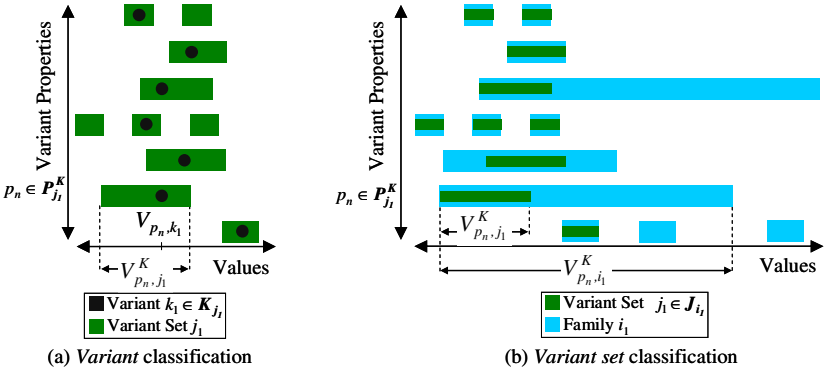
(a) **Property** categories (b) **Value Range** categories

Fig. 2. Property and **Restricted Value** specializations

As mentioned before, if a particular *variant* is a member of a given *variant set*, then each *variant property* must assume values belonging to the range specified by the *variant set*, as prescribed in (6).

$$\forall k \in \mathrm{K} : k \in \mathrm{K_j} \Rightarrow \forall\, p \in \mathrm{P_k}, \mathrm{V_{p,k}} \subseteq \mathrm{V^K_{p,j}} \tag{6}$$

where $\boldsymbol{V}_{p,k}$ is the set of values that *property p* assumes for *variant k* and $\boldsymbol{V}_{p,j}{}^{K}$ in the range of possible values for *property p* set by *variant set j*. Fig. 3a conceptualizes this idea.



(a) *Variant* classification (b) *Variant set* classification

Fig. 3. *Variant* and *Variant Set* classification conceptual notions

Similarly, if a given *variant set* is a member of a certain *family*, then, for each *variant property* pertaining to such *variant set*, the range of possible values must be included within the range fixed by the *family*. This specification is formalized in (7).

$$\forall\, j \in \mathrm{J} : j \in \mathrm{J_i} \Rightarrow \forall\, p \in \mathrm{P^K_j}, \mathrm{V^K_{p,j}} \subseteq \mathrm{V^K_{p,i}} \tag{7}$$

where $\boldsymbol{V}_{p,i}{}^{K}$ is the value range corresponding to *property p*, which is defined by *family i*. This notion is conceptually shown in Fig. 3b.

2.2 Product Unambiguous Definition

One basic assumption of the proposed model is that each product concept must be unambiguously identified. This would allow implementing proper classification

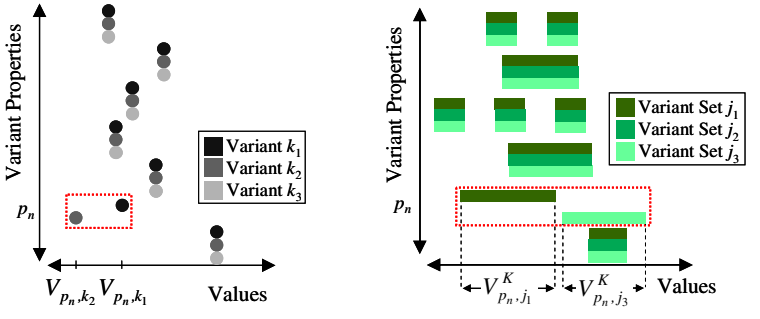
mechanisms. Thus, at the lower level of the hierarchy it is implied that all *variants* must be different. In the context of this approach two *variants* are considered distinct if either their sets of *properties* are different or there exists at least one *property* which assumes dissimilar values for each of these *variants*, as formalized in (8).

$$\forall k, k' \in \mathrm{K},\ k \neq k' \Leftrightarrow \mathrm{P_k} \neq \mathrm{P_{k'}} \vee \exists\, p \in \mathrm{P_k} : \mathrm{V_{p,k}} \neq \mathrm{V_{p,k'}} \tag{8}$$

Fig. 4(a) conceptualizes various *properties* and their corresponding values for three different *variants*. As it can be seen, $\boldsymbol{P}_{k_1}=\boldsymbol{P}_{k_2}\neq\boldsymbol{P}_{k_3}$ and $\boldsymbol{V}_{p_n,k_1}\neq\boldsymbol{V}_{p_n,k_2}$, therefore $k_1\neq k_2\neq k_3$. Likewise, all *variant sets* must be strictly different. This occurs if either they specify distinct subsets of *variant properties* or there exists at least one associated *variant property* for which the intersection between their corresponding value ranges is empty. This notion is formally specified in (9).

$$\forall j, j' \in \mathrm{J},\ j \neq j' \Leftrightarrow \mathrm{P_j^K} \neq \mathrm{P_{j'}^K} \vee \exists\, p \in \mathrm{P_j^K} : \mathrm{V_{p,j}^K} \cap \mathrm{V_{p,j'}^K} = \{\,\} \tag{9}$$

Fig. 4(b) conceptualizes several *variant properties* and their corresponding value ranges for three different *variants sets*. As it can be seen, $\boldsymbol{P}_{j_1}{}^{K}=\boldsymbol{P}_{j_3}{}^{K}\neq\boldsymbol{P}_{j_2}{}^{K}$ and $\boldsymbol{V}_{p_n,j_1}{}^{K}\neq\boldsymbol{V}_{p_n,j_3}{}^{K}$, therefore $j_1\neq j_2\neq j_3$.



(a) *Variant* uniqueness (b) *Variant Set* uniqueness

Fig. 4. *Variant* and *Variant Set* uniqueness schematic representations

Finally, *families* should be strictly different. This occurs if they specify distinct sets of *variant properties* or there exists at least one associated *variant property* for which the intersection of their corresponding value ranges is empty, as prescribed in (10).

$$\forall i, i' \in \mathrm{I},\ i \neq i' \Leftrightarrow \mathrm{L_i^K} \neq \mathrm{L_{i'}^K} \vee \exists\, p \in \mathrm{P_i^K} : \mathrm{V_{p,i}^K} \cap \mathrm{V_{p,i'}^K} = \{\,\} \tag{10}$$

Fig. 5 conceptually shows various *variant properties* and their corresponding value ranges for three different *families*. As it can be seen, $\boldsymbol{P}_{j_1}{}^{K}=\boldsymbol{P}_{j_3}{}^{K}\neq\boldsymbol{P}_{j_2}{}^{K}$ and $\boldsymbol{V}_{p_n,j_1}{}^{K}\neq\boldsymbol{V}_{p_n,j_3}{}^{K}$, therefore $j_1\neq j_2\neq j_3$. These assumptions assure an unambiguous classification criterion. In consequence, each *variant* should be a member of a unique *variant set* and each *variant set* should be a member of a unique *family*. Therefore, each *variant* would belong to a unique *family*.

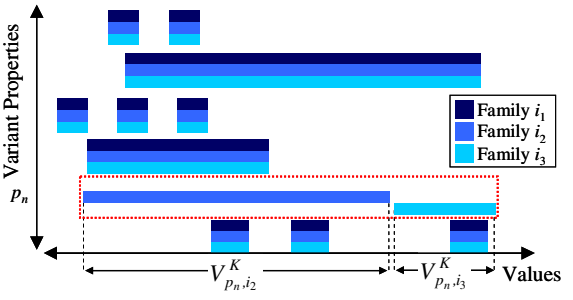


Fig. 5. *Family* uniqueness schematic representation

2.3 Product Structure

Another essential challenge of product modeling is the representation of product structures. Regarding this issue, *families* are classified into *compound* and *simple families*. In the first case, *compound families* can be decomposed into other *families* (a set of "parts" can be identified). On the other hand, *simple families* cannot be further decomposed. Along the same line of reasoning, *variant sets* and *variants* are classified into *compound* and *simple variant sets*, and *compound* and *simple variants*, respectively. These notions are formally stated in (11) and (12).

$$(a)\ \mathrm{I} = \mathrm{I}^{\mathrm{C}} \cup \mathrm{I}^{\mathrm{S}}\ ;(b)\ \mathrm{J} = \mathrm{J}^{\mathrm{C}} \cup \mathrm{J}^{\mathrm{S}}\ ;(c)\ \mathrm{K} = \mathrm{K}^{\mathrm{C}} \cup \mathrm{K}^{\mathrm{S}} \tag{11}$$

$$(a)\ \mathrm{I}^{\mathrm{C}} \cap \mathrm{I}^{\mathrm{S}} = \{\ \};(b)\ \mathrm{J}^{\mathrm{C}} \cap \mathrm{J}^{\mathrm{S}} = \{\ \};(c)\ \mathrm{K}^{\mathrm{C}} \cap \mathrm{K}^{\mathrm{S}} = \{\ \} \tag{12}$$

where $\mathbf{I}^{\mathrm{C}}/\mathbf{J}^{\mathrm{C}}/\mathbf{K}^{\mathrm{C}}$ is the subset of compound families/variant sets/variants and $\mathbf{I}^{\mathrm{S}}/\mathbf{J}^{\mathrm{S}}/\mathbf{K}^{\mathrm{S}}$ is the subset of simple families/variant sets/variants.

To keep model consistency, it is assumed that low-level *compound/simple abstractions* are members of high-level *compound/simple abstractions*. These assumptions are prescribed in (13) to (15).

$$(a)\ \forall k \in \mathrm{K}^{\mathrm{C}} : k \in \mathrm{K}_j \Rightarrow j \in \mathrm{J}^{\mathrm{C}}\ ;(b)\ \forall k \in \mathrm{K}^{\mathrm{S}} : k \in \mathrm{K}_j \Rightarrow j \in \mathrm{J}^{\mathrm{S}} \tag{13}$$

$$(a)\ \forall\, j \in \mathrm{J}^{\mathrm{C}} : j \in \mathrm{J}_i \Rightarrow i \in \mathrm{I}^{\mathrm{C}}\ ;(b)\ \forall\, j \in \mathrm{J}^{\mathrm{S}} : j \in \mathrm{J}_i \Rightarrow i \in \mathrm{I}^{\mathrm{S}} \tag{14}$$

$$(a)\ \forall k \in \mathrm{K}^{\mathrm{C}}\ \exists i \in \mathrm{I}^{\mathrm{C}} : k \in \mathrm{K}_{\mathrm{i}}\ ;(b)\ \forall k \in \mathrm{K}^{\mathrm{S}}\ \exists i \in \mathrm{I}^{\mathrm{S}} : k \in \mathrm{K}_{\mathrm{i}} \tag{15}$$

According to the proposal of Giménez et al. [3], *generic structures* are defined for *compound families*, which in turn can be modified by *compound variant sets* in order to allow the construction of particular BOMs for *compound variants*. Specifically, one or more *generic structures* are associated with each *compound family*. Then, each *variant set* specifies one *generic structure* of the corresponding *family* and from this particular one it derives the *structure* shared by all its members. Finally, actual BOMs (at the *variant* level) are obtained from the *structure* defined at *variant set* level.

Generic structures are classified into *composition* and *decomposition structures* depending on whether the *compound family* is composed of *generic components* (*families*) or it is decomposed into *generic derivatives* (*families*), as indicated in (16).

$$(a)\ \forall i \in \mathrm{I}^{\mathrm{C}}\ \exists s \in \mathrm{S} : s \in \mathrm{S}_{\mathrm{i}}\ ;\ (b)\ \mathrm{S} = \mathrm{S}^{\mathrm{C}} \cup \mathrm{S}^{\mathrm{D}}\ ;\ (c)\ \mathrm{S}^{\mathrm{C}} \cap \mathrm{S}^{\mathrm{D}} = \{\ \} \tag{16}$$

where $\boldsymbol{S}$ is the set of *structures*, $\boldsymbol{S}_i$ is the subset of *structures* associated with *family i*, and $\boldsymbol{S}^C/\boldsymbol{S}^D$ the subset of *composition/decomposition structures*.

The class diagram shown in Fig. 6 illustrates the specialization of the *family* concept (**Family**) into *compound* and *simple family* (**CFamily** and **SFamily**, respectively), the specialization of the *generic structure* concept (**Structure**) into *composition* and *decomposition structures* (**CStructure** and **DStructure**, respectively), and the definition of *generic structures* through *structural relations* (**StructuralRelation**), which are classified into *composition* and *decomposition structural relations* (**CStructuralRelation** and **CStructuralRelation**, respectively).

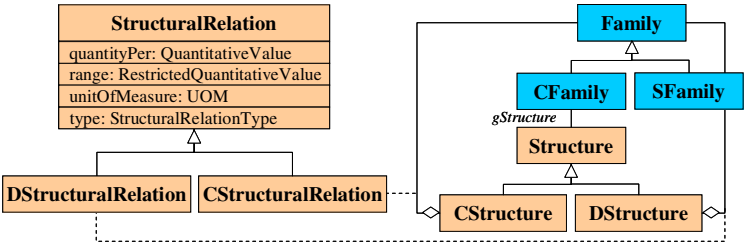


Fig. 6. *Generic structure* representation

A *structural relation* is established between the *generic structure* and the corresponding generic components/derivatives (*compound* or *simple families*). This relation provides information about the quantity of the generic component/derivative required/obtained per unit of *compound family*. In addition, the range of possible values for the quantity mentioned above, its unit of measure, and the relation type are also defined.

Three types of *structural relations* are adopted: *mandatory*, *optional* and *selective*. The chosen type determines whether a given *structural relation* can be removed from a *generic structure* by a *variant set*. Thus, when it is *mandatory*, the relation must exist; if it is *optional*, it can be eliminated; and when it is *selective*, only one relation of this type must be chosen (the other ones must be removed). See clauses (17)-(18).

$$(a)\ \forall s \in \mathrm{S}^{\mathrm{C}}\ \exists i \in \mathrm{I} : i \in \mathrm{I}_{\mathrm{s}}^{\mathrm{GC}}\ ;\ (b)\ \mathrm{I}_{\mathrm{s}}^{\mathrm{GC}} = \mathrm{I}_{\mathrm{s}}^{\mathrm{GCm}} \cup \mathrm{I}_{\mathrm{s}}^{\mathrm{GCo}} \cup \mathrm{I}_{\mathrm{s}}^{\mathrm{GCs}} \tag{17}$$

$$(a)\ \forall s \in \mathrm{S}^{\mathrm{D}}\ \exists i \in \mathrm{I} : i \in \mathrm{I}_{\mathrm{s}}^{\mathrm{GD}}\ ;\ (b)\ \mathrm{I}_{\mathrm{s}}^{\mathrm{GD}} = \mathrm{I}_{\mathrm{s}}^{\mathrm{GDm}} \cup \mathrm{I}_{\mathrm{s}}^{\mathrm{GDo}} \cup \mathrm{I}_{\mathrm{s}}^{\mathrm{GDs}} \tag{18}$$

where $\boldsymbol{I}_s^{GC}/\boldsymbol{I}_s^{GD}$ is the set of generic components/derivatives for *structure s*, $\boldsymbol{I}_s^{GCm}/\boldsymbol{I}_s^{GCo}/\boldsymbol{I}_s^{GCs}$ the subset of *mandatory/optional/selective* generic components for *structure s* and $\boldsymbol{I}_s^{GDm}/\boldsymbol{I}_s^{GDo}/\boldsymbol{I}_s^{GDs}$ the subset of *mandatory/optional/selective* generic derivatives for *structure s*. As mentioned before, a given *compound variant set* specifies one (and only one) of the *generic structures* associated with the *family* of which it is member, being able to eliminate some *structural relations*, but just those that are not *mandatory*. See clauses (19)-(23) representing these concepts.

$$(a)\forall\, j\in \mathrm{J}^{\mathrm{C}} : j\in \mathrm{J}_{\mathrm{i}}\ \exists\, s\in \mathrm{S}_{\mathrm{i}} : j\in \mathrm{J}_{\mathrm{s}}^{\mathrm{C}}; (b)\forall\, j\in \mathrm{J}^{\mathrm{C}}, j\in \mathrm{J}_{\mathrm{s}}^{\mathrm{C}} \wedge j\in \mathrm{J}_{\mathrm{s}'}^{\mathrm{C}} \Leftrightarrow s=s' \tag{19}$$

$$\forall\, j\in \mathrm{J}_{\mathrm{s}}^{\mathrm{C}} : s\in \mathrm{S}^{\mathrm{C}}\ \exists\, i\in \mathrm{I}_{\mathrm{s}}^{\mathrm{GC}} : i\in \mathrm{I}_{\mathrm{j}}^{\mathrm{GC}} \tag{20}$$

$$\forall\, j\in \mathrm{J}_{\mathrm{s}}^{\mathrm{C}} : s\in \mathrm{S}^{\mathrm{C}} \Rightarrow \mathrm{I}_{\mathrm{j}}^{\mathrm{GC}} \subseteq \mathrm{I}_{\mathrm{s}}^{\mathrm{GC}} \wedge \mathrm{I}_{\mathrm{s}}^{\mathrm{GCm}} \subseteq \mathrm{I}_{\mathrm{j}}^{\mathrm{GC}} \tag{21}$$

$$\forall\, j\in \mathrm{J}_{\mathrm{s}}^{\mathrm{C}} : s\in \mathrm{S}^{\mathrm{D}}\ \exists\, i\in \mathrm{I}_{\mathrm{s}}^{\mathrm{GD}} : i\in \mathrm{I}_{\mathrm{j}}^{\mathrm{GD}} \tag{22}$$

$$\forall\, j\in \mathrm{J}_{\mathrm{s}}^{\mathrm{C}} : s\in \mathrm{S}^{\mathrm{D}} \Rightarrow \mathrm{I}_{\mathrm{j}}^{\mathrm{GD}} \subseteq \mathrm{I}_{\mathrm{s}}^{\mathrm{GD}} \wedge \mathrm{I}_{\mathrm{s}}^{\mathrm{GDm}} \subseteq \mathrm{I}_{\mathrm{j}}^{\mathrm{GD}} \tag{23}$$

where $\boldsymbol{J}_s^{\,C}$ is the subset of *compound variant sets* whose *structure* derives from *generic structure s* and $\boldsymbol{I}_j^{\,GC}/\boldsymbol{I}_j^{\,GD}$ is the set of generic components/derivatives (*families*) from which the *variant sets* taking part in the *structure* of *variant set j* are selected.

Thus, a particular *variant set* must be selected for each non-eliminated generic component/derivative. In other words, for each *family* assuming the role of generic component/derivative in a non-eliminated *structural relation*, a *variant set* being member of such a *family* must be specified, as prescribed in (24).

$$(a)\ \forall\, i\in \mathrm{I}_{\mathrm{j}}^{\mathrm{GC}}\ \exists\ j'\in \mathrm{J}_{\mathrm{i}} : j'\in \mathrm{J}_{\mathrm{j}}^{\mathrm{GC}}\ ; (b)\ \forall\, i\in \mathrm{I}_{\mathrm{j}}^{\mathrm{GD}}\ \exists\ j'\in \mathrm{J}_{\mathrm{i}} : j'\in \mathrm{J}_{\mathrm{j}}^{\mathrm{GD}} \tag{24}$$

where $\boldsymbol{J}_j^{\,GC}/\boldsymbol{J}_j^{\,GD}$ is the set of components/derivatives of *variant set j*. Fig. 7 depicts the representation of the *structure* of a *compound variant set*.

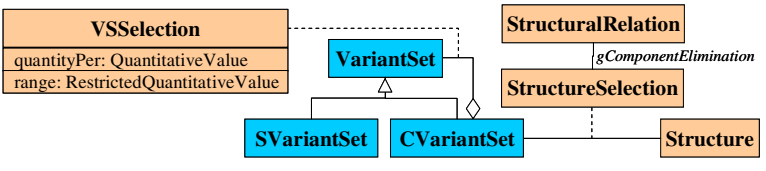


Fig. 7. Compound variant set structure representation

When a *variant set selection* is carried out, a new "quantity per" value and a new range of possible values can be specified. The condition to be satisfied is that the new range must be included within the range stipulated by the associated *structural relation*.

Finally, each *compound variant* adopts the *structure* defined by the *variant set* of which it is a member of, and specifies a particular member (*variant*) of each *variant set* assuming the generic component/derivative rol in such a *structure*. See (25)-(26).

$$k\in \mathrm{K}_{\mathrm{j}} : j\in \mathrm{J}_{\mathrm{s}}^{\mathrm{C}} \wedge s\in \mathrm{S}^{\mathrm{C}} \Rightarrow \forall\ j'\in \mathrm{J}_{\mathrm{j}}^{\mathrm{GC}}\ \exists\, k'\in \mathrm{K}_{\mathrm{j}'} : k'\in \mathrm{K}_{\mathrm{k}}^{\mathrm{C}} \tag{25}$$

$$k\in \mathrm{K}_{\mathrm{j}} \wedge j\in \mathrm{J}_{\mathrm{s}}^{\mathrm{C}} \wedge s\in \mathrm{S}^{\mathrm{D}} \Rightarrow \forall\ j'\in \mathrm{J}_{\mathrm{j}}^{\mathrm{GD}}\ \exists\, k'\in \mathrm{K}_{\mathrm{j}'} : k'\in \mathrm{K}_{\mathrm{k}}^{\mathrm{D}} \tag{26}$$

where $\boldsymbol{K}_k^{\,C}/\boldsymbol{K}_k^{\,D}$ is the set of components/derivatives of *variant k*. Fig. 8 shows the single-level BOM representation corresponding to a *compound variant*.

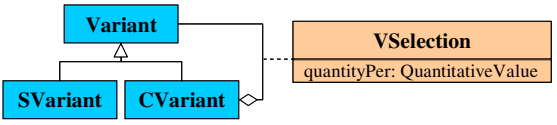


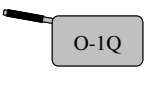
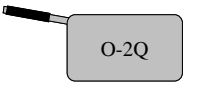
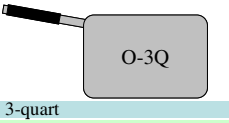
Fig. 8. *Variant* BOM representation

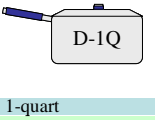
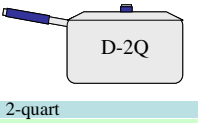
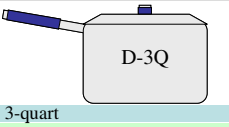
3 Case Study

In this section, the proposed approach is employed to represent the data associated with the set of cookware products illustrated in Fig. 9.

As it can be seen, the set of products corresponds to the *family* of saucepans. Three *variant sets* were identified: ordinary, deluxe and professional saucepans; each one grouping three *variants*. The first *variant set* represents the economical line of saucepans. They are characterized by having only one pan handle, not having a lid and being manufactured with medium-quality materials. The second *variant set* represents the intermediate product line. In this case, saucepans are characterized by

SAUCEPANS	**Intrinsic Properties** total demand: 200 total revenue: 8.68	**Variant Properties** size: {1-quart, 2-quart, 3-quart} lid?: {yes, no} steel line: {regular, clad}	 no. of pan handles: {1,2} lid handle shape: {loop, knob} handle line: {basic, executive}

ORDINARY SAUCEPANS			
Intrinsic Properties total demand: 120 total revenue: 4.44			
Variant Properties			
size: {1-quart, 2-quart, 3-quart}	1-quart	2-quart	3-quart
lid?: {no}	no	no	no
steel line: {regular}	regular	regular	regular
no. of pan handles: {1}	1	1	1
handles line: {basic}	basic	basic	basic

DELUXE SAUCEPANS			
Intrinsic Properties total demand: 60 total revenue: 3.00			
Variant Properties			
size: {1-quart, 2-quart, 3-quart}	1-quart	2-quart	3-quart
lid?: {yes}	yes	yes	yes
steel line: {clad}	clad	clad	clad
no. of pan handles: {1}	1	1	1
lid handle shape: {knob}	knob	knob	knob
handles line: {executive}	executive	executive	executive

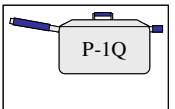
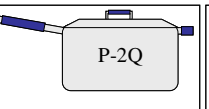
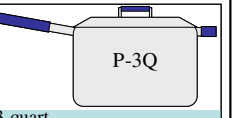
PROFESSIONAL SAUCEPANS			
Intrinsic Properties total demand: 20 total revenue: 1.24			
Variant Properties			
size: {1-quart, 2-quart, 3-quart}	1-quart	2-quart	3-quart
lid?: {yes}	yes	yes	yes
steel line: {clad}	clad	clad	clad
no. of pan handles: {2}	2	2	2
lid handle shape: {loop}	loop	loop	loop
handles line: {executive}	executive	executive	executive

Fig. 9. Set of products considered in the case study

having one pan handle, a lid with knob handle and being manufactured with high-quality materials. The last *variant set* represents the most complete line of products, which are recognized by having two pan handles, a lid with loop handle, apart from being manufactured with high-quality materials. *Variants* within a given *variant set* differ only in size. The three standardized sizes of saucepans are "one-quarter", "two-quarter" and "three-quarter". The *abstraction hierarchy* regarding this case study is depicted in Fig. 10.

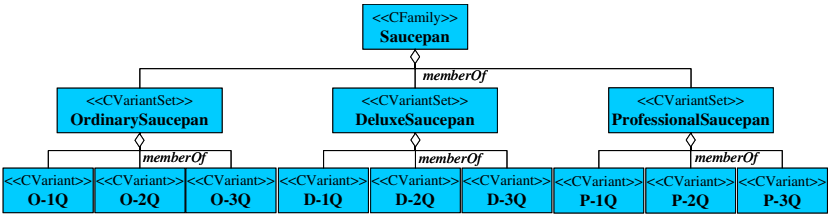


Fig. 10. *Abstraction hierarchy* associated with the case study

Clearly, *family*, *variant sets*, and *variants* are *compound abstractions,* since they are composed of other *abstractions* representing subassemblies. Fig. 9 also presents some of the *properties* associated with the different levels of abstraction. Two *intrinsic properties* were exemplified for the *family* of saucepans (*total demand* and *total revenue*) and six *variant properties* were included in such a *family* (*size*, *lid?*, *steel line*, *no. of pan handles*, *lid handle shape* and *handle line*). Besides, the same *intrinsic properties* were defined for all *variant sets* (i.e. *total demand* and *total revenue*).

In relation to *variant properties*, the *lid handle shape* one was eliminated from ordinary saucepans since they have no lids. Property values and value ranges are also shown in Fig. 9. Some examples of the concepts and relations presented in Figs. 1 to 3 are given in Figs. 11 and 12. Basically, Fig. 11 shows the definition of a specific *variant property* by a given *family* and the elimination of such a *variant property* by a particular *variant set*. It is also shown the narrowing of its value range. In turn, Fig. 12 illustrates examples of *intrinsic properties* associated with *product abstractions*.

From Fig. 9 it can be seen that the ranges specified for *variant properties* at the *variant set* level are comprised within the range stipulated for each *variant property* at the corresponding *family* level. Alike, values of *variant properties* specified at the *variant* level are comprised within the range established by the corresponding *variant set*.

On the other hand, it is verified that all *variant sets* are strictly different. Ordinary saucepans are different from deluxe and professional ones because the sets of *variant properties* are distinct. Despite possessing the same *variant properties*, deluxe and professional saucepans are also different because the value ranges defined for some *variant properties* have no common elements (values). For example, the number of pan handles is fixed to one for deluxe saucepans and to two for professional ones. Moreover, all *variants* are dissimilar since, within each *variant set,* they vary in size, as mentioned before.

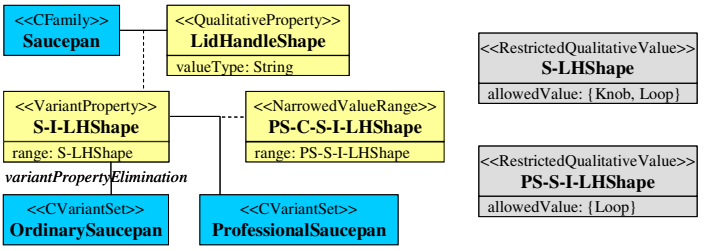


Fig. 11. *Variant property* definition

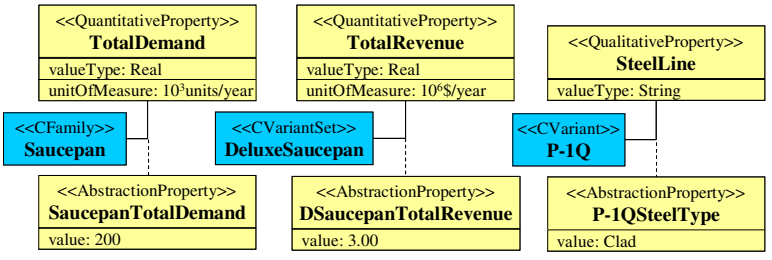


Fig. 12. Representation of *abstraction properties*

Regarding *product structures*, the *generic composition structure* associated with the *family* is represented in Fig. 13. Saucepans are generically composed of a pan assembly (*mandatory*) and a lid assembly (*optional*). In both cases the "quantity per" is exactly equal to 1.

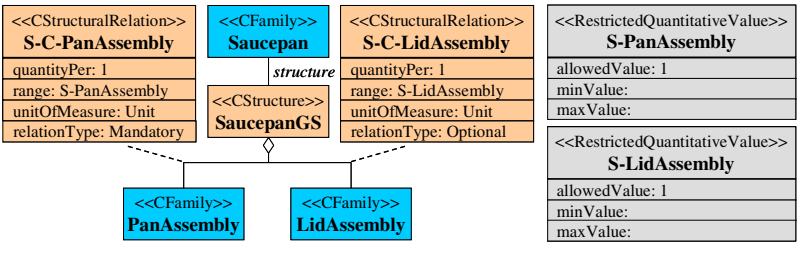
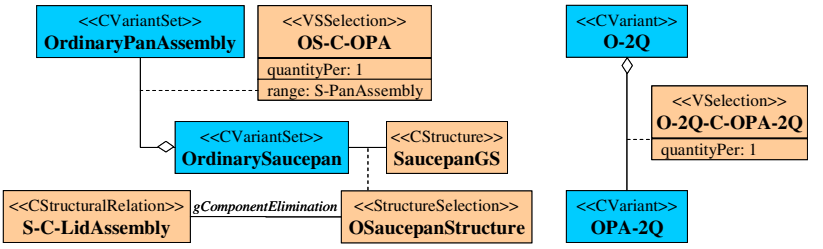


Fig. 13. Saucepans´ *generic structure*

An example of a *variant set structure* is shown in Fig. 14(a). In this case, ordinary saucepans are composed of only ordinary pan assemblies. The *family* of lid assemblies was eliminated from the *generic structure*. In turn, the set of ordinary pan assemblies was selected as a generic component. An example of a *variant* single-level BOM is depicted in Fig. 14(b). As it is shown, a two-quarter ordinary saucepan is composed of a two-quarter ordinary pan assembly.



(a) Ordinary saucepans *structure* (b) 2-Q ordinary saucepan BOM

Fig. 14. Examples of the *Variant Set structure* and *Variant* BOM generation concepts

4 Final Remarks

In this paper, a novel model for product data management is presented. The proposal is based on a three-level abstraction hierarchy. It differs from similar approaches because an unambiguous criterion to classify product concepts along the hierarchy is defined. At the same time, it provides foundations to handle data (dis)aggregation processes in a systematic way. Moreover, the proposed model attempts to offer more expressiveness, flexibility and reuse of information at the different levels of abstraction.

Regarding model validation, a prototype of a Distributed Product Data Management (DPDM) system that supports the classification criterion described in Section 2 is currently under development. Several case studies of different complexity are being addressed in order to validate the conceptual model and the classification procedure as well as to evaluate its practical applicability. Preliminary results show that many of the complexities associated with the management of massive product data can be effectively tackled by implementing this novel hierarchical product-property model.

Acknowledgements. This work has been supported by CONICET, UTN, and UNL.

References

1. De Lit, P., Danloy, J., Delchambre, A., Henrioud, J.-M.: An Assembly-Oriented Product Family Representation for Integrated Design. IEEE Transactions on Robotics and Automation 19, 75–88 (2003)
2. Du, X., Jiao, J., Tseng, M.M.: Architecture of Product Family: Fundamentals and Methodology. Concurrent Engineering: Research and Application 9, 309–325 (2001)
3. Giménez, D.M., Vegetti, M., Henning, G.P., Leone, H.P.: PRoduct ONTOlogy: Defining product-related concepts for logistics planning activities. Computers in Industry 59, 231–241 (2008)
4. Yan, W., Chen, S.-H., Huang, Y., Mi, W.: A data-mining approach for product conceptualization in a web-based architecture. Computers in Industry 60, 21–34 (2009)

Collaborative, Participative and Interactive Enterprise Modeling

Joseph Barjis

Delft University of Technology, Jaffalaan 5, 2628 BX Delft, The Netherlands
j.barjis@tudelft.nl

Abstract. Enterprise modeling is a daunting task to be carried out from a single perspective. A challenge to this whole complexity is conflicting descriptions given by different actors when business processes are documented. Often enterprise modeling takes rounds of iterations and clarification before the models are verified and validated. In order to expedite the modeling process and validity of the models, in this paper we propose an approach called collaborative, participative, and interactive modeling (CPI Modeling). The main objective of the CPI approach is to furnish an extended participation of actors that have valuable insight into the enterprise operations and business processes. Achieving this goal with any modeling method and language could be quite challenging. For CPI Modeling to succeed the modeling method should adhere to certain qualities. Next to the CPI Modeling approach, this paper discusses an enterprise modeling method that is simple, and yet powerful to capture intricate enterprise processes and simulate them.

Keywords: CPI Modeling, Collaborative modeling, Interactive modeling, Participative modeling, Business process modeling, Business process simulation, Enterprise modeling, Enterprise simulation, DEMO methodology, Language-action perspective, Petri net.

1 Introduction

Enterprise modeling is a daunting task and usually error-prone process. The problem is not the maturity of the modelers and analysts involved, but the complex socio-technical nature of an enterprise. In particular, enterprise modeling in its broader context encompasses business processes, human and organizational issues, technical aspects such as information systems and enabling IT applications. The enterprise modeling challenge could be best seen in the definition given in [14], which is defined as "computational representation of the structure, activities, processes, information, resources, people, behavior, goals and constraints of business, government, or other enterprise". Basically, this definition alone suffices to persuade the reader that a traditional way of enterprise modeling will not yield too much success if non-innovative approaches and methods deployed.

A traditional approach towards enterprise modeling, especially enterprise processes modeling is to delegate the work to analysts and modelers who will be normally visiting the enterprise under study, read the existing documentation, conduct a series

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 651–662, 2009.
© Springer-Verlag Berlin Heidelberg 2009

of interviews, after which enterprise models are developed by the modelers alone. Only several back and forth visitation and refinements will allow for a more complete model of the enterprise to emerge. This approach has a number of problems. As the interaction of the analysts with the enterprise employees become more and more often, the enterprise becomes more reluctant and less interested to allocate their most needed human resources to be involved in the project, which will be seen as waste of time. This is the reason that many enterprises will consider the modeling part as hindering the IS development project. In turn, modelers, not having sufficient rounds of iteration, will end up with a model that is either incomplete, or there are many assumptions that are intuitively made by the molders. As a result, the model may contain a lot of flaws. These flaws remain quite undetected as majority of enterprise process modeling is not based on formal semantics to check the models and simulate their dynamic behavior.

A comprehensive enterprise modeling requires that the models have to capture three phenomena such as enterprise processes, enterprise business rules, and enterprise information, which should be integrated in the corresponding information system deliverable [23]. Thus, it is very important that enterprise models are accurate and complete as flawed models will result in inadequate final systems, especially that system developers may rely on the models for developing an actual system as many of these complex systems are already driven by models – model-driven system development.

Traditional approaches of enterprise modeling, where modelers and analysts are the players and the rest (process owners, mangers, stakeholders, experts) are either passive participants or even absent from the scene, are least likely to result in accurate models. Firstly, the modern enterprise business processes are too complex to be understood, captured and documented by modelers and analysts alone. Secondly, because the resultant models should be approved by all stakeholders and decision makers, and only then the models can be implemented.

In order to address theses challenges in enterprise modeling, innovative approaches have been discussed and introduced such as participative enterprise modeling [22, 27]. A central goal of enterprise modeling is to discover domain knowledge and document the enterprise existing business processes. The role of participative modeling is to represent this knowledge in a coherent and comprehensive model, create shared understanding, consolidate different stakeholder views, and in order to do so an extended participation of stakeholders is crucial [25].

From the foregoing, it becomes obvious that participative enterprise modeling is rather a necessity than a choice. However, two other things that need to be considered for a successful participation are collaboration and interaction. We refer to this approach as collaborative, participative, and interactive modeling (CPI Modeling). It is imperative that each of the three notions is given explicit attention, which we will do in the following section.

2 Collaborative-Participative-Interactive Enterprise Modeling

In this section we will explain why the emphasis of this work is specifically on ‘collaboration’, ‘participation’, and ‘interaction’ as three constituents of success for complete, accurate, acceptable, and expedited enterprise modeling.

2.1 Collaboration

- Collaboration has been proven to result in a fast and accurate model development when modeling of complex phenomenon of socio-technical nature is involved. But collaboration itself is a subject of engineering approach, where explicit scenarios and guidelines are required to be designed for facilitating collaboration.
- Collaboration also requires guidance and orchestration by an experienced facilitator. Facilitators with modeling experience and basic domain knowledge are more successful to lead the participants.
- Even a more extended collaboration of interdisciplinary nature (analysts, consultants, IT professional, social researchers) would be required to cope with complex design objects and propose innovative solutions.
- Finally, collaboration requires that modelers (with knowledge of modeling language and techniques) and analysts (with expertise in analysis and modeling) are collaborating to design accurate models.

2.2 Participation

- Often, the goal of enterprise is to document the operations and business processes, and create shared understanding of what and how an enterprise is operating. This shared understanding, which is achieved through a complete picture of the enterprise business processes, would be difficult without participation of key employees of units involved by implication.
- Often, also enterprise models convey different accounts of business processes from different unit's perspectives. Numerous iterations of enterprise modeling are required to build an accurate model, which makes the process tedious, extensive and costly. Consolidation of different accounts and expedition of enterprise modeling are another challenge that is hard to cope with, unless the modeling involves participation of process owners.
- Often, the ultimate deliverables of enterprise modeling are changes in the current practice, organizational restructuring, or investment in new technology. It will be hard to achieve these ultimate goals without an extended participation and approval of stakeholders.
- Finally, verification and validation of complex models have always been challenging. Presence of the process owners and business units managers and their participation in the modeling will result in immediate verification and validation of the models, especially when simulation methods are deployed in the process, which demonstrate enactment of the models.

2.3 Interaction

- Innovative tools and technologies are needed to furnish interaction of modelers, analysts, and participants of enterprise modeling. While technologies such as large interactive smart boards (including the one that can be shared remotely) are creating interactive environment, these are specific tools that furnish the success.
- Tools should be intuitive, easy to follow, and powerful to capture complex interactions.

- Static models are no longer sufficient to create the shared understanding that complex enterprise process models should accomplish. Tools that simulate the processes allow to capture dynamic behavior of the constructed models, observe the effects of changes, and manipulate the models.

These three aspects constitute the so called CPI Modeling approach, where each aspect is a dimension: the collaboration aspect represent the *Experts* (analysts) dimension; the participation aspect represents the *Users* (stakeholders) dimension; and the interaction aspect represents the *Technology* (tools) dimension. However, these constituents comprise only the approach we adapted for enterprise modeling. The next challenge is the method, language, and notations used for enterprise modeling. The question is can we use UML and IDEF with the same level of success? Is there any advantage of using EPC over UML? Or none of these methodologies are suitable to support the CPI Modeling approach that requires interactive modeling with different participants onboard (from mature business analysts and professional molders to employees that have no technical or modeling background). In the following section, we briefly discuss why some enterprise modeling methodologies may not be suitable.

3 The Modeling Method Consideration

Depending on specific situations and contexts, enterprise analysts can develop a prototype of the envisioned system, and study their behavior [4]; the analyst can develop mathematical models [3, 15] and abstractions of systems and study them by calculating output parameters of the models; they can draw static pictures using diagrams and then study the diagrams such as IDEF [16, 21], UML [7, 17, 26], EPC [11], Petri Nets [2], etc. Each of these approaches presents certain benefits and, of course, certain limitations and drawbacks. However, diagrammatic representation of models represents enormous interests and practical value as it poses least cognitive load and great communication capability [18, 19]. In selecting a diagrammatic method, it is important that the chosen approach (method and tool) adequately fits the problem situation. It is extremely important for the system's ultimate success to ensure the quality of the modeling methods and tools [6].

As discussed in [20], there are certain quality attributes that a modeling method should adhere to such as syntactic, semantic, and pragmatic qualities. Syntactic qualities require rules and grammar that drive modeling and prevent construction errors. In pragmatic qualities, strong emphasis is put on executability of models, their visualization, simulation and animation. Adding to these qualities, in fact, CPI modeling implies participation of non-modelers and non-analysts, therefore ease of use, natural compliance with the way the organization conducts its operations, and intuitiveness require a huge extent of model simplicity. Model simplicity means that the modeling method should be easy and simple for understanding and construction, and yet powerful to capture the complexity of underlying situation.

For example, UML is difficult to learn, it is complex, and there seem to be too many diagrams [24]. For that reason, not all of the UML diagrams are used by analysts [10]. Therefore, to use UML as a modeling language in CPI approach will pose certain challenges. Moreover, an enterprise is a social environment where human

actors naturally interact while requesting actions or committing themselves to certain actions. Therefore capturing these social characteristics will definitely surface the requirements for an enterprise modeling method.

Simulation of models in a CPI environment will add a lot of value for sharing understanding. Therefore, the modeling language should lend to simulation and allow to check the models consistency and completeness. We propose that Petri net possesses balanced properties (expressivity, intuitiveness, formal semantics) to serve the purpose. Most of the conventional models are checked and analyzed via translation to other formal diagrams using mapping procedures. For instance, UML activity diagrams are often translated to Petri nets for checking [12, 13]. Another widely accepted method, investigated in [8], is Event-driven Process Chain (EPC). The authors propose a 5-step guideline to translate EPC models to Petri net models in order to investigate whether the process is correctly described in EPC. The analysis showed that ambiguities of EPC models will result in faulty Petri net executions. Finally, IDEF diagrams are also semi-formal diagrams that present little pragmatic value in collaborative and participative modeling environment, where simulation of the models is very important.

As for the modeling notations that compete with Petri net, e.g. BPMN, EPCs, Role-Activity-Diagrams, IDEF, UML, RIVA etc., Petri net is known for rigorous semantics, logics and formalism, and also is widespread among researchers, practitioners and a variety of academic disciplines. In addition, Petri net is supported by a large number of tools for its analysis. In [1], author identifies three main reasons why Petri net possesses advantageous features: formal semantics despite the comprehensive graphical representation; state-based representation instead of event-based; abundance of analysis techniques. Process modeling techniques ranging from informal techniques (e.g., dataflow diagrams) to formal ones (e.g., process algebra) are event-based, while Petri net approach allows state-based modeling.

The enterprise modeling method we propose is based on the DEMO transaction concept (developed for social systems) and Petri net graphical notations lending to simulation.

4 DEMO Transaction

In this section we first briefly discuss the DEMO transaction's original diagram and concept, and then we introduce a diagram, which is based on Petri net semantics. We use Petri net graphical notations to allow the resulting model to be simulated.

4.1 Original Notations

This section is based on the original works conducted in the framework of DEMO methodology. The results of more than a decade development of this methodology are summarized in [9]. DEMO is an acronym for Design and Engineering Methodology for Organizations (see www.demo.nl for more information).

According to the DEMO theory, social actors in organization perform two kinds of acts: *production act* (P-acts, for short) and *coordination act*s (C-acts, for short). By engaging in P-acts, the actors bring about new results or facts, e.g., they deliver service or produce goods. Examples of P-acts are: register a student into new course;

issue a ticket for a show; make a payment. By engaging in C-acts, the actors enter into communication, negotiation, or commitment towards each other. Examples of C-acts are: making a request for new course; presenting an issued ticket to the customer. The generic pattern in which the two kinds of actions (P-acts and C-acts) occur is called *transaction*, see Figure 1. In fact, a transaction is steps of **C-act→P-act→C-act** that correspondingly result in C-fact (e.g., commitment to register a student) and P-fact (e.g., do register a student).

A transaction is carried out in three phases: the *Order* phase (O-phase, for short), the *Execution* phase (E-phase, for short) and the *Result* phase (R-phase, for short). These three phases involve two actor roles. The actor role that initiates a transaction is called *initiator*. The actor role that carries out a production act is called *executor*.

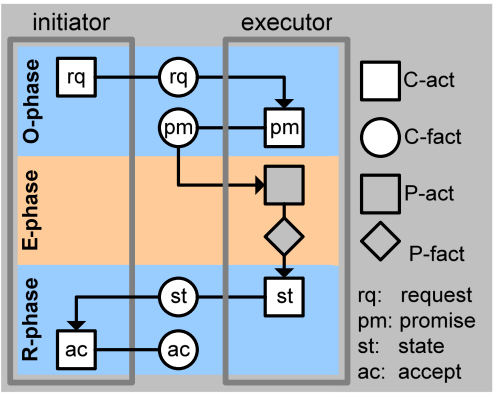


Fig. 1. Basic Transaction Concept (adapted from Dietz 2006)

In the following sub-section we introduce and discuss the extensions we made to the DEMO transaction diagram based on Petri net. In the terminologies used, we return to earlier terms used for P-acts and C-acts, i.e., instead of P-act, we refer to it as *action*, and C-act as *interaction*.

4.2 Extended Notations

A business transaction is a pattern of action and interaction. An *action* is the core of a business transaction and represents an activity that brings about a new result. An *interaction* is a communicative act involving two actor roles to coordinate or negotiate a particular action.

Each business transaction is carried out in three distinct phases (see Figure 2a):

- *Order phase (O)*, during which an actor makes a 'request' for a service or good towards another actor. This phase represents a number of communicative acts or interactions. This phase ends with a commitment ('promise') made by the second actor, who will deliver the requested service or good.
- *Execution phase (E)*, during which the second actor fulfills its commitment, i.e., 'produce' the service or good. This phase represents a productive act.

- *Result phase (R)*, during which the second actor does 'present' the first actor with the service or good prepared. This phase also represents a number of communicative acts or interactions. This phase ends with the 'accept' of the service or good by the first actor.

These phases are abbreviated as O, E and R correspondingly (see Figure 2b). The figure illustrates a business transaction in detailed generic form and simple OER form. Note that the order (O) and result (R) phases are interactions and the execution (E) phase is an action.

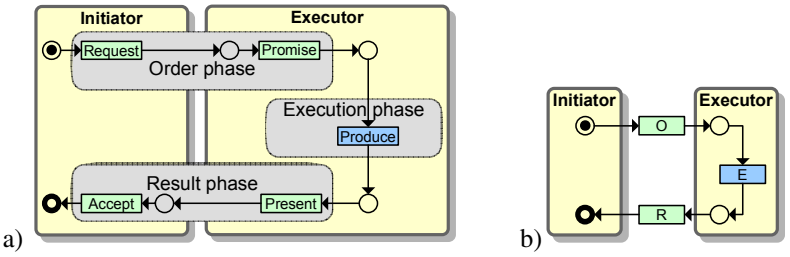


Fig. 2. Business transaction: a) a generic form, b) an OER form

In a structured language, a transaction is described according to Table 1, where a transaction is portrayed through the *activity pattern* it represents (e.g., placing an order), its *initiator* (e.g., customer), *executor* (e.g., supplier), and the *result* it delivers (e.g., a new order is created).

Table 1. Transaction description in a structured language

Transaction:	**Atomic process (e.g., *placing an order)***
Initiator	Name of the role that initiates the transaction (e.g., *customer)*
Executor	Name of the role that executes the transaction (e.g., *supplier)*
Result	The result created as the transaction is carried out (e.g., *a new order is created)*

Now that we have discussed the CPI Modeling approach and the modeling method supporting it, we introduce a case study where we tested both the approach and the modeling method. However, we have to skip a whole set of notations and modeling constructs we have developed that can be used as building block or components in enterprise modeling. The interested reader is referred to [5] for more reading on modeling notations and constructs developed based on the DEMO transaction concept and Petri net graphical semantics.

5 Case Study: DutchPlast BV Enterprise

This case study was conducted on DutchPlast BV, a plastic production company, located in Westland-Area, The Netherlands. The company recently launched an

initiative to review its business processes and improve the current processes with reducing delays, and developing new information systems that will support the redesigned business processes.

5.1 CPI Enterprise Modeling

In the CPI Modeling approach we applied for conducting enterprise modeling of DutchPlast, we pretty much followed the recommendations for participative modeling suggested in [25]. However, we put strong emphasis on interactive modeling, tools and technology. We organized a half day session with the following participants:

- DutchPlast – technical director and order and procurement director – business process owners with high expertise and authority about the enterprise operations.
- Enterprise modeling expert – an author of enterprise modeling methodology
- Modelers – expert modelers.
- Facilitator – a professional collaboration facilitator with PhD and expertise in using interactive smart-boards
- Observers – a group of observers to document the session.
- Graduate students – a group of graduate students who are graduated from an enterprise modeling course.

The modeling session was conducted on a large interactive smart-board allowing use of electronic color pens on a touch-screen. The modeling covered the whole enterprise business processes.

The end of the session feedback and discussion allowed to learn about the importance of simple and intuitive modeling notations. What was the most important feedback received is that the participants would prefer a simulation model over static diagrammatic models, although we have not conducted the simulation part in the session. The processes that we modeled are described in the following sub-section.

5.2 The Customer Order Process

It should be noted that the description presented in this section is significantly reduced for this paper. Accordingly, the number of transactions and the enterprise processes model are also reduced.

A customer wants DutchPlast to produce a product. He can send an e-mail or fax with a short description of the desired product to a salesperson. The customer can also contact the salesperson directly by telephone. The salesperson needs to have as much information (colour, measurements, etc.) as needed about the product. The salesperson uses this information in a calculation program to estimate the costs. Also the delivery time is estimated based on the planning in the information system. A salesman creates an offer with the costs and delivery date. The offer is sent to the customer. The customer may decline the offer or accept.

The order can either be a standard product or a customized product. A standard product is easy to produce as information for production is already available. A customized product requires designing and work preparation. For customized products, the technical designer sketches the product based on the information in the order. If the product cannot be designed because information is missing or incorrect

Table 2. DutchPlast business transactions

T1:	**Making an offer**
Initiator	Customer
Executor	Salesman
T2:	**Placing an order**
Initiator	Customer
Executor	Salesman
T3:	**Produce the product**
Initiator	Salesman
Executor	Product producer
T4:	**Check the product quality**
Initiator	Salesman
Executor	Production manager
T5:	**Internal transportation**
Initiator	Salesman
Executor	Internal transporter
T6:	**External transportation**
Initiator	Salesman
Executor	External transporter
T7:	**Pay external transportation**
Initiator	External transporter
Executor	Salesman
T8:	**Pay the order**
Initiator	Salesman
Executor	Customer
T9:	**Report to collection agency**
Initiator	Salesman
Executor	Collection agency
T10:	**Recurring stock control**
Initiator	Stock controller
Executor	Stock controller
T11:	**Restocking**
Initiator	Salesman
Executor	Supplier
T12:	**Pay for restocking**
Initiator	Supplier
Executor	Salesman
T13:	**Production preparation**
Initiator	Salesman
Executor	Production manager
T14:	**Create technical drawing**
Initiator	Production manager
Executor	Technical drawer
T15:	**Customer approval**
Initiator	Salesman
Executor	Customer
T16:	**Production planning**
Initiator	Salesman
Executor	Production planner
T17:	**Order required materials**
Initiator	Salesman
Executor	Supplier
T18:	**Pay for the materials**
Initiator	Supplier
Executor	Salesman

in the order, he contacts the customer to supply the needed information. Also new materials will need to be ordered if a customized product requires it.

A standard product is produced using an open mould technique. After preparing everything for production the product is ready to be assembled. The producer will assemble the product.

When a product has been produced, it's checked by the production manager for quality purpose. Only then transportation of the product is arranged. When the delivery is within the Westland-area, the transportation is being done by the internal transportation department of DutchPlast. Otherwise, it will be outsourced to an external transport company. In that case, DutchPlast also has to pay the external transport company.

On a monthly basis, the financial administrator checks for unpaid bills. He will then contact the client or contact a collection agency to handle the debtor.

In the following section, based on the above description, we try to identify business transactions according to the DEMO transaction concept.

5.3 DutchPlast Business Transactions

The business transactions contained in Table 2 were identified in a collaborative manner with participation of experts, business process owners and business analysts using the description of business process of the previous section.

This list of transactions served as an input for the enterprise model construction. A complete model of the DutchPlast enterprise was developed in an interactive manner. Since the model itself is not in the scope of this paper, we did not include the model into this paper, however, through the list of business transactions included in the table and the related actors (units) we want to convey the whole complexity that an enterprise model may present. The purpose for this case study was to apply the CPI Modeling approach and reflect on the experience. Some conclusions are drawn in the following section where we discuss the CPI Modeling experience.

6 Conclusions

In this paper we have discussed the CPI Modeling approach and its importance for enterprise modeling, especially enterprise business processes. We also discussed that CPI approach can be only useful if it is supported by a suitable enterprise modeling method and language that is simple, and yet powerful.

As the DutchPlast case study revealed, an enterprise process model can be very complex. Although we omitted almost half of the case, still the processes are complex enough to be captured without collaboration of business modelers, analysts, facilitator and participation of process owners and managers. The CPI modeling approach not only helps with the production of enterprise models and expedites the modeling process, but it allows validation and verification of the model almost immediately. To this end, both the participation and presence of the business owners and simulation of the model can help (in the DutchPlast case we skipped simulation due to time limitation).

Application of suitable methodology is crucial to communicate among the participants. A method that consists of a moderate number of elements, intuitive notations, and closely resembles the natural way of working in an enterprise has significant advantage as the participants do not have modeling background. A small set of elements reduces cognitive load of the participants not familiar with the modeling method. The modeling method should fit into the social sitting of an enterprise, which is a social system by virtue.

The discussed case study is a starting point for more extended research. The CPI Modeling approach opens up an opportunity for many potential directions, into which the CPI modeling approach can be developed. First of all, we intend to create a rich simulation environment around this approach and method allowing simulation of enterprise processes with animative features including building a library of customized entities. This will create a more realistic replica of the enterprise under study. Another interesting research topic would be to develop scenario based guidelines that will allow practitioners to apply CPI approach without extensive involvement of a facilitator. One more interesting observation that also leads to some research was that the participants did not fully collaborate at the beginning of the CPI Modeling session. This means that fro a full collaboration of the participants certain procedures and measures should be developed.

Acknowledgements. It is a pleasure to thank several people who participated in the session: first of all, the author of the DEMO Methodology for his participation in the session as an expert; the DutchPlast senior employees (commercial director, technical director); the facilitator of the session; colleagues and graduate students who helped with observing the session.

References

1. van der Aalst, W.M.P.: Three Good Reasons for Using a Petri-net-based Workflow Management System. In: Proceedings of the International Working Conference on Information and Process Integration in Enterprises (1996)
2. van der Aalst, W.M.P., Desel, J., Oberweis, A. (eds.): Business Process Management: Models, Techniques and Empirical Studies. Springer, Heidelberg (1998)
3. Aris, R.: Mathematical Modelling Techniques, New York (1994)
4. Arnowitz, J., Arent, M., Berger, N.: Effective Prototyping for Software Makers. Morgan Kaufmann, Elsevier, Inc. (2007)
5. Barjis, J.: The Importance of Business Process Modeling in Software Systems Design. Journal of The Science of Computer Programming 71(1), 73–87 (2008)
6. Bollojy, N., Leung, S.S.K.: Assisting Novice Analysts in Developing Quality Conceptual Models with UML. Communications of the ACM 49(7) (July 2006)
7. Booch, G., Rumbaugh, J., Jacobson, I.: The Unified Modelling Language User Guide. Addison-Wesley, Reading (1999)
8. Dehnert, J., van der Aalst, W.M.P.: Bridging the Gap Between Business Models and Workflow Specifications. Int. Journal of Cooperative Information Systems 13(3), 289–332 (2004)
9. Dietz, J.L.G.: Enterprise Ontology –Theory and Methodology. Springer, Heidelberg (2006)

10. Dobing, B., Parsons, J.: How UML is used. Communications of the ACM 49(5) (May 2006)
11. van Dongen, B.F., van der Aalst, W.M.P., Verbeek, H.M.W.: Verification of EPCs: Using Reduction Rules and Petri Nets. In: Proceedings of the 17th Conference on Advanced Information Systems Engineering (2005)
12. Eichner, C., Fleischhack, H., Meyer, R., Schrimpf, U., Stehno, C.: Compositional Semantics for UML 2.0 Sequence Diagrams Using Petri Nets. In: SDL Forum 2005, pp. 133–148 (2005)
13. Eshuis, R.: Symbolic Model Checking of UML Activity Diagram. ACM Transactions on Software Engineering and Methodology 15(1) (January 2006)
14. Fox, M.S., Gruninger, M.: On Ontologies and Enterprise Modelling. In: International Conference on Enterprise Integration Modelling Technology 1997 (1997)
15. Gershenfeld, N.: The Nature of Mathematical Modeling. Cambridge University Press, Cambridge (1998)
16. IDEF, Family of Methods web page (2003), http://www.idef.com
17. Jacobson, I., Booch, G., Rumbaugh, J.: The Unified Software Development Process. Addison Wesley Longman (1998)
18. Koehler, J., Hauser, R., Küster, J., Ryndina, K., Vanhatalo, J., Wahler, M.: The role of visual modeling and model transformation in business-driven development. ENTCS, vol. V, p. 211 (2008)
19. Larkin, J.H., Simon, H.A.: Why a diagram is (sometimes) worth ten thousand words. Cognitive Science 11(1), 65–100 (1987)
20. Lindland, O., Sindre, G., Sølvberg, A.: Understanding quality in conceptual modeling. IEEE Software 11(2), 42–49 (1994)
21. Mayer, R.J., Painter, M., deWitte, P.: IDEF Family of Methods for Concurrent Engineering and Business Re-engineering Applications. Knowledge Based Systems, Inc. (1992)
22. Persson, A.: Enterprise Modelling in Practice: Situational Factors and their Influence on Adopting a Participative Approach, PhD dissertation, Stockholm University (2001)
23. Prakash, N.: Bringing Enterprise Business Processes into Information System Products. In: Stirna, J., Persson, A. (eds.) The Practice of Enterprise Modeling. LNBIP, vol. 15 (2008)
24. Siau, K., Cao, Q.: Unified Modeling Language (UML) - a complexity analysis. Journal of Database Management 12(1), 26–34 (2001)
25. Stirna, J., Persson, A., Sandkuhl, K.: Participative Enterprise Modeling: Experiences and Recommendations. In: Krogstie, J., Opdahl, A.L., Sindre, G. (eds.) CAiSE 2007 and WES 2007. LNCS, vol. 4495, pp. 546–560. Springer, Heidelberg (2007)
26. Torchiano, M., Bruno, G.: Enterprise modeling by means of UML instance models. SIGSOFT Software Engineering Notes 28(2) (2003)
27. de Vreede, G.-J.: Participative Modelling for Understanding: Facilitating Organizational Change with GSS. In: Proceedings of the 29th HICSS (1996)

Part IV

Software Agents and Internet Computing

e-Learning in Logistics Cost Accounting Automatic Generation and Marking of Exercises

Markus Siepermann[1] and Christoph Siepermann[2]

[1] Technische Universität Dortmund, Department of Business Information Management
44227 Dortmund, Germany
Markus.Siepermann@uni-dortmund.de
[2] University of Kassel, Department of Production and Logistics
34109 Kassel, Germany
Siepermann@wirtschaft.uni-kassel.de

Abstract. This paper presents the concept and realisation of an e-learning tool that provides predefined or automatically generated exercises concerning logistics cost accounting. Stu-dents may practise where and whenever they like to via the Internet. Their solutions are marked automatically by the tool while considering consecutive faults and without any intervention of lecturers.

Keywords: Automatic marking, e-learning environment, online practicing, randomly-generated exercises, logistics, cost accounting.

1 Denotation

Due to the internationalisation of trading and markets in the past decade the role of logistics has become more and more important. Logistics in business education evolved from a branch of production to an independent subject. Many new degree programmes specialise in logistics whose graduates are much sought-after by enterprises.

A fundamental topic of logistics education is logistics cost accounting. As logistics costs make up 10-25% of industrial enterprises' total costs [1] or 5-10% of turnover [2], [3] and that logistics services have the utmost importance for differentiation in competition [4], it is essential for an enterprise to be able to calculate the costs of logistics services and to allocate these costs accurately to the relevant cost units that generated the costs. Unfortunately, traditional cost accounting is not able to perform these tasks in a satisfactory way. Therefore, several publications are engaged with the development of approaches to overcome the deficits of traditional cost accounting concerning the treatment of logistics costs or the costs of a firm's service areas in general. In the following we want to show how e-learning techniques can support lessons in logistics cost accounting by generating sophisticated exercises that help students to understand and to practise the different methods of logistics cost accounting proposed in literature.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 665–676, 2009.
© Springer-Verlag Berlin Heidelberg 2009

2 Logistics Cost Accounting

As traditional cost accounting mainly focuses on production, a source-related allocation of logistics costs to cost units cannot be achieved. The reason is that logistics costs are mostly overhead costs. Traditional cost accounting now argues, that those overhead costs suffer from a missing connection to products and therefore can only be allocated to cost units via value-based allocation measures like direct material costs, direct wages and production costs. Thus, the higher direct material costs, direct wages and production costs of a product are, the more this product is charged with logistics costs, neglecting the real use of logistics services. For example, within traditional cost accounting, a product that is composed of many low cost parts will hardly be charged with logistics costs, although it causes much more procurement costs than a product composed of only a few but expensive parts. The same situation holds for production logistics costs. Applying machine hours as allocation measures supposes a coherence between the production time of a product and its claim for production logistics services. But, rather, the complexity of the production processes is the appointing determinant. Finally, distribution logistics costs are not determined by production costs (as implicitly assumed by traditional cost accounting), but by a product's storage and transport attributes, e.g. dimension, weight etc. [5]. Taking into account the significance of logistics costs and services pointed out above, these faults in cost allocation may lead to fatal errors in product-related decision-making due to wrong information concerning a product's logistics costs.

Another lack of traditional cost accounting can be seen in the fact that logistics costs are not reported separately in product costing, but as a part of procurement, production, sales and administration overhead costs (lack of transparency). For these reasons, basically two approaches have been developed or can be applied in order to achieve a source-related allocation of logistics costs to cost units:

– Weber proposes a refinement of traditional cost accounting [5]. A similar approach is provided by Reichmann [6]. Both approaches can be applied as absorption accounting or marginal costing.

– The second alternative consists of applying activity-based costing (ABC), which was explicitly developed for indirect service types and therefore can be assumed to be suitable for logistics cost accounting as well. But ABC originally was designed as full-absorption accounting [7], [8], which implies that it does not separate costs into its fixed and variable parts. Therefore, it is not able to provide any information with regard to short-term decision-making, such as accepting or refusing an additional order. Certain further developments in ABC try to remedy this deficiency by separating costs according to their dependency on the operating level and/or convertibility in time, in addition to the separation between costs for process-volume-induced and process-volume-neutral activities introduced by Horváth/Mayer [7]. In this context, we especially have to mention the approaches by Reichmann/Fröhling [9], Glaser [10], Mayer [11] and Dierkes [12]. It is striking that all these further developments in ABC based on direct costing come from the German-speaking world. The reason for this is that direct costing systems are very advanced in Germany thanks to the works of Kilger [13] and Riebel [14] and German academics therefore attach great importance to an appropriate cost splitting.

Although each approach is different from the others they have the following in common:

– In all of these approaches, logistics cost centers are defined as final cost centers.
– Except for the Reichmann approach, which only defines logistics specific surcharge rates, all approaches try to allocate logistics costs of cost centers to cost units via transfer rates that are based on volume-based allocation measures. This implies that we can clearly identify a relation between the output of logistics cost centers and the usage of this output by cost units. If not, we have to do without the allocation of the respective logistics costs or value-based surcharge rates have to be used as a remedy.

The fundamental difference between the approaches according to Reichmann and Weber on the one hand and the activity-based costing approaches on the other hand consists of the different number of calculation steps: Within the Reichmann and Weber approach, the costs of logistics cost centers are immediately allocated to the cost units via cost center-based allocation measures. By contrast, activity-based costing firstly allocates logistics costs to activities. In a second step, these activity costs are allocated to the cost units via activity-based allocation measures. Activities can be aggregated hierarchically over several levels. Commonly there are two hierarchy levels.

The further developments in activity-based costing mainly differ in the way of cost splitting. Cost splitting can refer to the costs' dependency on the operating level, which leads to the differentiation between variable and fixed costs, whereby fixed costs additionally can be differentiated according to the readiness to operate, and/or to the costs' convertibility in time (i.e. their commitment period), which leads to the differentiation between costs that are degradable in the short-, medium- and long-term.

3 Logistics Cost Accounting in e-Learning

Now, how can we support teaching logistics cost accounting appropriately by e-learning techniques? To answer this question, we firstly have to distinguish three key parts of university education:

– Teaching
– Practicing
– Assignment and Grading

Concerning the teaching task, students mostly prefer traditional lectures [15], even if they are criticized for being antiquated, and they don’t want them to be replaced by electronic lectures [16]. Therefore, e-learning should not replace traditional lectures, but rather serve as an additional feature that assists traditional lectures with interactivity and multi-media elements. Because of this, we will focus on the practicing and assignment task where the student’s individuality is on top. In order to do justice to that individuality, we have to deal individually with each student and his or her abilities and deficits. That is, where e-learning systems offer a great benefit because of the uneven ratio between students and lecturers. Without the help of an e-learning system we cannot occupy with the individual abilities and solutions of

students [17]. But e-learning allows us to provide exercises that are suitable to the individual situation and knowledge of each student, to mark the students' solutions automatically and to give individual hints concerning the lessons a student ought to revise again. The Internet offers numerous possibilities to realise such individual e-learning exercises and tools so that students are able to practice whenever and wherever they wish. Such a self-steered learning is one of the most efficient paths to comprehension [18].

But the e-learning exercises have to be interactive [19] and sophisticated enough so that students have to find the answer on their own by using the learned approaches and their own knowledge. Therefore, e-learning exercises should not only be composed of simple forms like multiple choice, true-false questions, jumbled sentences or fill-in-the-blank [20]. In these cases, the practising students don't really need their knowledge because often they easily can guess the correct answers by systematically reducing the number of possible answers [21].

Didactically good exercises that really help students understand the contents of lectures should not contain the answer and the problem-solving process in a more or less apparent form. Students should rather be forced to prove their abilities to solve a problem. That can be done by not only evaluating the final result of an exercise, but also by considering the chosen way of problem-solving [17]. In this context, it is important that students can choose their own problem-solving process without any restrictions. Restrictions should only appear if there are technical reasons [22].

Unfortunately, such interactive and sophisticated exercises either do not exist or are very rarely supported by e-learning systems because of their complexity [17]. In most cases today, those exercises are still corrected by human beings [23], [24], [21].

But the disadvantages of traditional exercises are obvious: The manual marking of exercises absorbes resources and results in a time delay between practising, marking and feedback about mistakes and lessons to repeat although immediate feedback would be very valuable [25], [26]. To remedy these problems, automatically marked exercises are needed [27], [28].

Due to the various degrees of freedom, this task is quite difficult to accomplish because often there is not only one correct answer, but rather several answers that are more or less correct. Thus, students' solutions cannot only be classified in the two categories right or wrong. We can instead identify a scale of correctness concerning the solutions because of consecutive faults and several more or less correct ways to solve the problem [29].

Now, logistics cost accounting consists of a mathematical calculus with several calculation steps. To perform these calculation steps, we need a set of variables to be calculated and combined in a determined way, but the order of the calculation steps is (up to a certain degree) freely definable. Thus, the calculation steps can be represented in a calculation grid where the vertices are the variables and functions and the edges are the relations between them. Because of this exercise structure, it is possible to generate exercises with random values and to automatically mark students' solutions.

In the following, we will present the concrete e-learning system that provides those exercises for logistics cost accounting. It allows lecturers to manually predefine exercises as well as to generate them automatically. The system automatically marks the students' solutions without intervention of a lecturer. The marked exercises are presented to the students with hints about their deficits and lessons to be repeated.

4 E-Learning Concept

4.1 Overview

Altogether, we can distinguish seven different approaches of logistics cost accounting: The approaches according to Weber and Reichmann, the original form of activity-based costing developed by Horváth and Mayer and the further developments in ABC developed by Reichmann and Fröhling, Glaser, Mayer and Dierkes. An overview of the principles of each approach is given by [30] and [31].

In order to be able to compare the results of these seven approaches to those of traditional cost accounting, the latter is also implemented in the e-learning system. This helps to demonstrate the mistakes of traditional cost accounting in allocating logistics costs to the products and therefore the need of a special logistics cost accounting system.

The major task of each logistics cost accounting systems consists of allocating a manufacturer's logistics costs properly (i.e. according to the products' claim of logistics services) to the cost units and in providing transparent information about the composition of these (product-related) logistics costs. Therefore, the subject of each exercise in logistics cost accounting is the calculation of product costs with special regard to logistics costs.

The relevant approaches and therefore the exercises that shall be implemented differ in

- the allocation measures used,
- the manner of cost splitting,
- the logistics cost categories (resulting from cost splitting) which are allocated to the product units and which are not, and
- the applied calculation scheme.

All types of exercises have in common the master data concerning cost types, cost centers, allocation measures, activities (if needed) and the logistical attributes of the products to be calculated. These master data are independent of the chosen type of exercise and can easily be extended if necessary.

When practising, each type of exercise comprises the following three steps:

- Calculating cost type-based, cost center-based and/or activity-based allocation measures,
- Calculating the product-related values of the logistical allocation measures (i.e. the activity coefficients)
- Performing product costing by using the results of steps one and two.

4.2 Generation of Exercises

The e-learning system provides two different ways to generate exercises: They can either be created manually by lecturers, or they can be generated automatically by the e-learning system. When *manually* creating an exercise, we have to choose the exercise type (i.e. the logistics cost accounting approach to be applied), the mattering cost types, cost centers, allocation measures, activities (if needed) and the (logistical) attributes of the products to be calculated (for an example of the latter see figure 1).

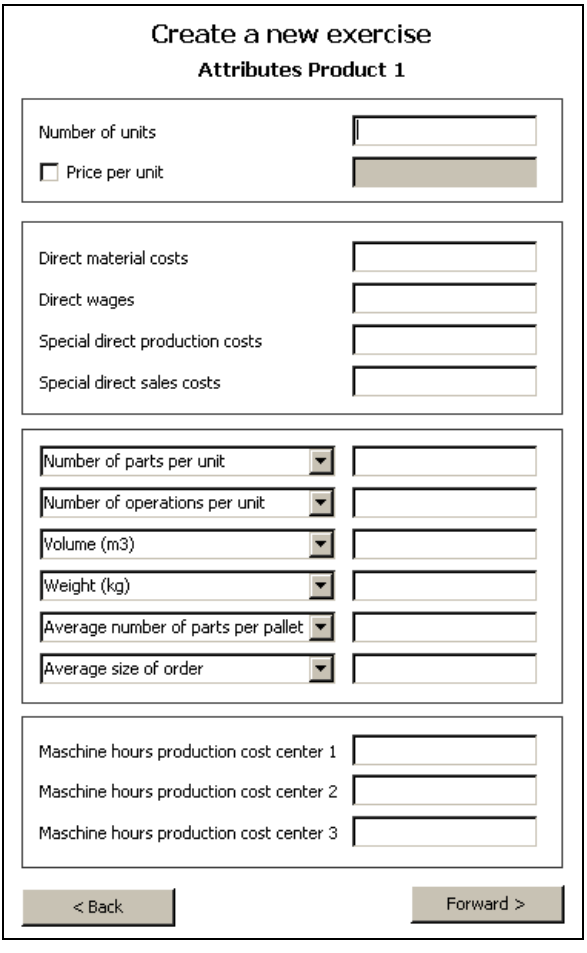


Fig. 1. User interface for lecturers

If the master data pool isn't sufficient, new master data can be added to the pool by lecturers. Based on the chosen master data, the e-learning system creates empty data sheets for the initial data that can be filled by lecturers with appropriate values. The calculation can either be done manually by the lecturer in order to test the exercise – the system then checks the solution and shows potentially made mistakes – or it can be performed automatically by the system. Finally, the lecturer can revise the exercise and make some changes before saving it.

When *automatically* generating an exercise, there are several interdependencies between the different data of exercises that have to be considered. For example, a realistic ratio between staff costs and material expenses, variable and fixed costs, direct and overhead costs should be guaranteed as well as a realistic level of the transfer and surcharge rates. These interdependencies are stored as rules in a rule database and are used each time an exercise is generated. The e-learning system

randomly generates the basic data and then adjusts these data by using the rules. The dependent and derived values are computed.

4.3 Difficulty Levels

In order to achieve a broad acceptance of e-learning, exercises should be suited to the actual knowledge of students [22]. For this reason, different difficulty levels should be offered. The possibility of choosing different underlying approaches leads to different difficulty levels and handling times, yet. Additional difficulty factors are the number of cost centers, activities, activity types (e.g. output-based and non-output-based ones), the kind of cost splitting etc. These elements are parameterised and build the fundamentals in computing difficulty level and target time of an exercise.

The number of cost centers or activities, for example, affects the number of calculation steps and therefore must have an impact on the target time. By contrast, the content-oriented difficulty level is influenced by the kind of cost splitting or the number of different activity types occurring in the exercise. Thus, the target time is computed with the number of calculation steps, and the difficulty level is computed with the help of the different difficulty parameters.

When automatically generating an exercise, the components of an exercise are chosen with respect to the given difficulty level. The target time is then also influenced by the chosen components: Complex components lead to longer target time, easy components can be handled faster.

Another variation of difficulty can be derived from the different conditions that may hold at the beginning of an exercise. At easy levels the whole calculation scheme can be provided to users. In this case, students only need to compute the correct values. At top levels, no presettings are made and students will have to design the problem solution process, as well as to compute the correct values.

4.4 Practising with Exercises

Normally, no presettings are made to students. Necessary variables have to be defined by students themselves. Therefore, variables can be created by defining two inputs: The name of the variable and the values. The name of the variable can either be chosen out of a list of possible names or it can be defined in a free text field. In order to recognise free user inputs, a fault-tolerant word recognition function, based upon well known metrics like the Levenshtein or the Damerau distance, is implemented. With the help of these metrics, the correct variable name can be identified out of the user's input.

In exam mode, an exercise has to be solved within the target time. When the target time expires, the solution is sent to the e-learning system and is automatically marked. Beyond the exam mode, a message that the normal target time has expired is displayed. After finishing, the total time of practising is compared to the target time and a grading is presented to the user.

4.5 Automatic Marking

As already mentioned above, each type of exercise is based upon a more or less simple form of calculation that can be modelled by calculation rules [32].

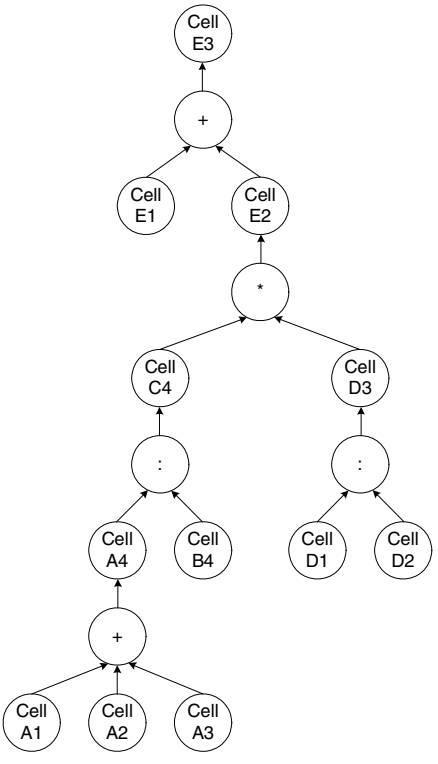


Fig. 2. Example of a calculation grid

Because of the free user input and the various possibilities to solve the problem provided by an exercise, a simple marking by comparing the results of the students' calculations with the reference solution won't succeed. Rather, we have to follow the students' solution processes in order to understand how they have reached their results. Therefore, we firstly recognise the used variables of the students' solutions. After that two calculation grids are reconstructed by using the rule database: The calculation grid of the reference solution and the student's calculation grid (see figure 2).

In this way, it is not only possible to compare the final result of the student's solution to the reference solution, but also to check whether the student has used a correct problem solving process. This can be done by comparing the two calculation grids step by step. Doing so, three kinds of faults can occur:

1. Necessary variables are missing.
2. Values of variables are faulty.
3. Needless variables are used.

If values are faulty the system marks the mistakes and inserts a hint. At first sight, the usage of needless variables is not critical. But it may happen that those needless variables aren't needless: They might be missing variables that haven't been recognised correctly. In order to avoid this marking mistake, not only the names but

also the values of the variables are used while recognising the variables. If a variable is missing, a percentage is subtracted from the total achievable score and the fault is marked. If there are further calculation steps, the marking algorithm proceeds despite missing or faulty variables. Doing so, we have to care about consecutive faults that result from missing or faulty values.

In order to recognise these consecutive faults, the marking procedure uses the already marked values and recalculates the following calculation steps with these faulty values. Thus, correct calculations are recognised as correct with respect to faulty values. Consecutive faults are also marked, but they don't lead to a reduction in scoring.

5 System Architecture

The following objectives were the major focus during the design of the e-learning system:

- Provision of predefined exercises
- Provision of randomly generated exercises
- Automatic marking of students' solutions
- Provision of an exam mode

In order to put these objectives into practise, the system requires an exercise administration module, a user administration module, an exercise generation module, a master database and a rule database, a configuration module, a content module and a representation module (see figure 3).

The *exercise administration module* stores and manages the exercises predefined by lecturers as well as the automatically generated ones. All exercises are classified according to their difficulty level. The difficulty level results from the different accounting methods and the calculation components which occur in an exercise. The *user administration module* manages every single user. Every user can act in different roles: Students can practice with exercises, automatically generate exercises and have a look at the marking of their solutions. Lecturers can predefine exercises or automatically generate them, work on exercises like students and have a look at students' solutions to gain an insight into student's knowledge. Administrators assign roles to each user, work on fundamental system parameters and adjust the parameterisation of the difficulty levels. The solutions of users are also stored in the user administration module.

The *exercise generation module* can be used by students and lecturers. Students can choose a difficulty level and choose whether they want to work on that exercise in the exam modus or not. Lecturers as well can automatically generate exercises and select a difficulty level. Additionally, they can in- or exclude single parameters to create a more specialised exercise. The generation module provides an exercise according to the chosen preferences and calculates the target time for solving with regard to the difficulty parameters. The exercise is generated with respect to the master data stored in the *master database* and to the rules stored in the *rule database*. The lecturers are able to add master data and rules to these databases via the *configuration module*.

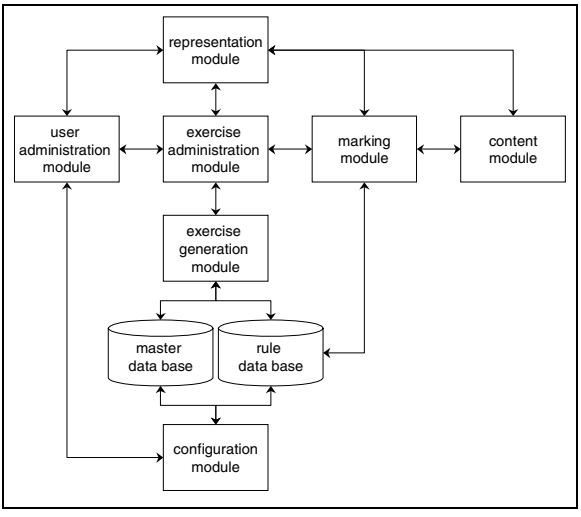


Fig. 3. Architecture of the e-learning system

After the expiration of the target time (if the exam modus was chosen) or after the exercise has been finished by the student, the solution is sent to the *marking module*.

This module evaluates the solution using the rule database, marks right and wrong elements and gives hints as to which lessons should be repeated via the *content module*. The exercise is represented via the *representation module*.

The presented e-learning system is a client-server-based system, developed with classical web technologies. Work on exercises and therefore exercise representation takes place at the client. The frontends of the administrative modules also run at the client side. All other modules are only operated on the server side.

6 Conclusions

In this paper we presented an e-learning system that generates and provides exercises concerning logistics cost accounting. As there are several approaches that can and have to be taken into consideration, logistics cost accounting offers various possibilities of practicing. The benefit of the system consists of the following advantages:

- Students can practise whenever and wherever they wish.
- Students get feedback in a predictable time due to automatic marking.
- Exercises are suitable to the students' actual individual knowledge.
- Innumerable exercises can be created automatically.
- Lecturers are relieved of routine jobs.

The system is now ready to use. Future work will focus on generalizing the system to an e-learning system for all types of cost accounting.

References

1. Schulte, C.: Logistik, 4th edn. Vahlen, München (2005)
2. European Logistics Association (ELA), A.T. Kearney Management Consultants: Differentiation for Performance – Excellence in Logistics 2004. Deutscher Verkehrs-Verlag, Hamburg (2004)
3. Pfohl, H.-C.: Logistiksysteme, 7th edn. Springer, Berlin (2004)
4. Wildemann, H.: Der Wertbeitrag der Logistik. Logistik Management 6(3), 67–75 (2004)
5. Weber, J.: Logistikkostenrechnung, 2nd edn. Springer, Berlin (2002)
6. Reichmann, T.: Controlling mit Kennzahlen und Management-Tools, 7th edn. Vahlen, München (2006)
7. Horváth, P., Mayer, R.: Prozeßkostenrechnung: Der neue Weg zu mehr Kostentransparenz und wirkungsvolleren Unternehmensstrategien. Controlling 1(4), 214–219 (1989)
8. Horváth, P., Mayer, R.: Konzeption und Entwicklungen der Prozeßkostenrechnung. In: Männel, W. (ed.) Prozeßkostenrechnung, pp. 59–85. Gabler, Wiesbaden (1995)
9. Reichmann, T., Fröhling, O.: Integration von Prozeßkostenrechnung und Fixkostenmanagement. Kostenrechnungspraxis 37(2), 63–73 (1993)
10. Glaser, K.: Prozeßorientierte Deckungsbeitragsrechnung. Vahlen, München (1998)
11. Mayer, R.: Kapazitätskostenrechnung. Vahlen, München (1998)
12. Dierkes, S.: Planung und Kontrolle von Prozeßkosten. DUV, Wiesbaden (1998)
13. Kilger, W.: Flexible Plankostenrechnung und Deckungsbeitragsrechnung, 10th edn. Gabler, Wiesbaden (1993)
14. Riebel, P.: Einzelkosten- und Deckungsbeitragsrechnung, 7th edn. Gabler, Wiesbaden (1994)
15. Glowalla, U., et al.: Verbessern von Vorlesungen durch E-Learning Komponenten. i-com 3(2), 57–62 (2004)
16. Bruns, H.: How to choose the right eLearning technique? Overview and recommendations. In: Bruns, H., Ambrosi, G.M. (eds.) eLearning and Economics, pp. 17–25. Books on Demand, Norderstedt (2002)
17. Siepermann, M.: Lecture Accompanying E-Learning Exercises with Automatic Marking. In: Proceedings of E-Learn 2005, pp. 1750–1755. Association for the Advancement of Computing in Education, Chesapeake (2005)
18. Kerres, M., Jechle, T.: Didaktische Konzeption des Telelernens. In: Issing, L.J., Klimsa, P. (eds.) Information und Lernen mit Multimedia und Internet, pp. 267–281. BeltzPVU, Weinheim (2002)
19. Haack, J.: Interaktivität als Zeichen von Multimedia und Hypermedia. In: Issing, L.J., Klimsa, P. (eds.) Information und Lernen mit Multimedia und Internet, pp. 127–136. BeltzPVU, Weinheim (2002)
20. Weidenmann, B.: Multicodierung und Multimodalität im Lernprozeß. In: Issing, L.J., Klimsa, P. (eds.) Information und Lernen mit Multimedia und Internet, pp. 45–62. BeltzPVU, Weinheim (2002)
21. König, M.: E-Learning und Management von technischem Wissen in einer webbasierten Informationsumgebung. Druckerei Duennbier, Duisburg (2001)
22. Lackes, R., Siepermann, M.: Bru-N-O'Mat – Automatically Generating and Marking Net Requirements Calculation Exercises. In: Proceedings of E-Learn 2007, pp. 7198–7204. Association for the Advancement of Computing in Education, Chesapeake (2007)
23. Schlageter, G., Feldmann, B.: E-Learning im Hochschulbereich: der Weg zu lernzentrierten Bildungssystemen. In: Issing, L.J., Klimsa, P. (eds.) Information und Lernen mit Multimedia und Internet, pp. 347–357. BeltzPVU, Weinheim (2002)

24. Kwan, R., Chan, C., Lui, A.: Reaching an Itopia in distance learning – A case study. AACE Journal 12(2), 171–187 (2004)
25. Kobi, E.E.: Lernen und Lehren, Haupt. Bern and Stuttgart (1975)
26. Strzebkowski, R., Kleeberg, N.: Interaktivität und Präsentation als Komponenten multimedialer Lernanwendungen. In: Issing, L.J., Klimsa, P. (eds.) Information und Lernen mit Multimedia und Internet, pp. 229–245. BeltzPVU, Weinheim (2002)
27. Bolliger, D., Martingale, T.: Key Factors for Determining Student Satisfaction in Online Courses. International Journal on E-Learning 3(3), 61–67 (2004)
28. Issing, L.J.: Instruktions-Design für Multimedia. In: Issing, L.J., Klimsa, P. (eds.) Information und Lernen mit Multimedia und Internet, pp. 151–176. BeltzPVU, Weinheim (2002)
29. Siepermann, M., Lackes, R.: Self-Generating and Automatic Marking of Exercises in Production Planning. In: Proceedings of the IADIS International Conference WWW/Internet 2007, vol. II, pp. 13–17 (2007)
30. Siepermann, C.: Fallstudie zur Logistikkostenrechnung: Darstellung und vergleichende Analyse verschiedener Verfahren. In: Günther, H.O., Mattfeld, D.C., Suhl, L. (eds.) Supply Chain Management und Logistik, pp. 291–316. Physika, Heidelberg (2005)
31. Siepermann, C.: Logistics Cost Accounting: Which Approach is Preferable? In: Logistics Bridges on Supply Chain, Proceedings of the 5th International Logistics & Supply Chain Congress 2007 in Istanbul, pp. 307–314 (2007)
32. Patel, A.: Kinshuk: Intelligent Tutoring Tools – A problem solving framework for learning and assessment. In: Proceedings of 1996 Frontiers in Education Conference – Technology-Based Re-Engineering Education, pp. 140–144 (1996)

Towards Successful Virtual Communities

Julien Subercaze[1], Christo El Morr[2], Pierre Maret[3], Adrien Joly[4], Matti Koivisto[5], Panayotis Antoniadis[6], and Masayuki Ihara[7]

[1] Université de Lyon, LIRIS UMR 5205, France
julien.subercaze@liris.cnrs.fr
[2] School of Health Policy and Management, York University, Canada
elmorr@yorku.ca
[3] Université de Lyon, LHC UMR 5516, France
pierre.maret@univ-st-etienne.fr
[4] Alcatel-Lucent Bell Labs, France
adrien.joly@alcatel-lucent.fr
[5] Mikkeli University of Applied Sciences, Finland
matti.koivisto@mikkeliamk.fi
[6] Universit Pierre et Marie Curie - Paris 6, France
Panayotis.Antoniadis@lip6.fr
[7] NTT COMWARE Corporation, Japan
ihara@acm.org

Abstract. With the multiplication of communication medium, the increasing multi-partner global organizations,the remote working tendencies,dynamic teams, pervasive or ubiquitous computing Virtual Communities (VCs) are playing an increasing role in social organizations currently and will probably change profoundly the way people interact in the future. In this paper, we present our position on the key characteristics that are imperative to provide a successful VC as well as the future directions in terms of research, development and implementation. We identify three main aspects (business, techniques and social) and analyze for each of them the different components and their relationships.

Keywords: Virtual Communities, VCs, Guidelines, Characteristics, Business model, Techniques, Social dimension.

1 Introduction

In the 1990's and due to the internet phenomena, a particular kind of communities was born: the Online Community, also known as Virtual Communities (VCs). VCs have been the subject of particular attention; it has been defined and classified in different ways [1,2,3].

VCs varied in the technologies they use (e.g., email lists, forums, chat rooms), the wide domain of applications (e.g. tourism, health, sociability, leisure). In the 1990's mobility emerged in the telecommunication industry and had a remarkable impact of VC research; particularly on design, infrastructures to use [4,5], services to offer[6], the user interface[7,8], the security [9] and the privacy of users.

With the multiplication of communication medium, the increasing multi-partner global organizations, the remote working tendencies, the dynamic teams, the pervasive

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 677–688, 2009.
© Springer-Verlag Berlin Heidelberg 2009

and wearable computing, VCs are meant to play an central role in social organizations and will profoundly change the way people interact. On the other hand, the emergence of the information society depends in a great part on the way information is exchanged between collaborating groups. In this context, VCs appear to be one of the pillars of the information society. The concept of community appears as a key feature for the development of tomorrows' applications in the information society. Nevertheless, there is a need to define the technologies, methodologies, and tools for the collection, management, exchange and use of information within communities as well as for the engineering of VC applications, in this regard practical engineering challenges are to be met. In this paper, we present our position on the key imperative characteristics to provide a successful VC, and on the future directions in terms of research, development and implementation. Researchers consider VCs from different aspects depending on their background and research prospects. Business professionals look into a profitable viable business model, while engineers search for the most efficient way to maintain connectivity, performance, the way information is organized and communicated; knowledge in VC is an attraction for engineers as well as information systems researchers. The ease of use of the VC through appropriate interface is another aspect of research and development. Decision making and social characteristics are important too. In the myriad of possible VC applications, finding the right set of characteristics that make a VC "work" is a challenge. In the following, we categorize the different characteristics that we believe are central for a successful VC. We identify three main aspects: business, technical and social dimensions. For each of these aspects, we will analyze the different components and their relationships.

2 Business Dimension

2.1 Startup Cost

Starting a VC does not necessarily require large investment in the early stage. The famous Facebook started as a hobby project and then expanded thanks to venture capital investors.. In the traditional curve of traffic in a successful community, there is a critical point where the traffic explodes very quickly. To be able to manage the traffic increase, the company should be able to answer the expectations of users while quickly installing new servers and keeping (or increasing) the quality of service in terms of response time. This is a critical phase because these investments are linked to periods of unavailability of the community, where users may be discourage if the unavailability is too long compared to the visible new advances. At this time most of the communities are not able to support such a heavy investment with their own funds. The examples of MySpace and Facebook show that the successful expansion was only possible trough external capital (Facebook example: $500,000 in the first round, and $12.8 Million one year later). In May 2008, Facebook took out $100 million in debt to invest in servers and increase headcount.

2.2 The Media Factor

The role of the media in the success of a community is also important. In the early stage of a community's lifecycle, the word of mouth, buzz, and the internet presence are the

most success factors. To get the needed attention, we identified three key success factors:(1)Propose innovative and high usable features for users,(2) Open your community to developers,(3) Promote the current technological standards, like RSS, web 2.0, usability rules.

In terms of media strategy, it is important to open a blog to promote your company, to keep the media/blogger communities aware of the latest news and advances of your community, and communicate often about the increasing number of members. Getting the attention of the mainstream media actors, like newspaper or television, is a sign of the success of a community. Recent examples of MySpace and Facebook showed that those networks reached their maximum of popularity just after the mainstream media shed the light on them. Traditional media have a low influence on the success of VCs since they will discover them once they are already successful, but they will give the last momentum to reach the peak. Viral marketing is a key factor of success on the internet. Successful projects like hi5, linkedIn, Facebook have reached their current dominant position without advertising and big marketing communication, and they then switched to traditional communication methods like news releases, interviews, conferences and advertising in mainstream media.

2.3 Only Actor or First Mover in the Market

"The world is too small for both of us". This quote summarizes well the rough battle waged by VCs. The number of internet users increases continuously but there is still no place for two VCs in the same field. Being the first actor to move is a key of success factor or at least helps to maintain a dominant position. A German clone of Facebook, Studivz, starting in 2005 is still maintaining its first position as social network for students in German speaking countries in spite of the rise of Facebook in Europe since 2007. The inertia of community members is very high; they migrate massively to competitor services only if they have substantial advantages that justify it. Nevertheless, if two competing communities cannot survive, each of them can focus on different segments of the market.

2.4 Attracting Users and Developing Loyalty

In the era of the so-called "web2.0" people are invited to contribute to websites and socialize. YouTube and Flickr are two of the most successful VCs that focus more on content sharing and allow to users to publish and consume user-generated content, instead of publisher provided, content. Besides, there are communities, which focus more on the social networking activity; in this domain, two VCs have had outstanding popularity: MySpace and Facebook. MySpace was one of the first social networks on the internet, allowing people and bands to create their own profile page which they can decorate with their favorite images, videos and music. People can also send private messages to each other, but public messages are usually preferred. The most common usage of MySpace is to create a page with as much customized content and friends as possible, in order to increase its visible popularity (i.e. the number of friends). It is a way for youngsters to boost their social ego, and for bands to spread their music virally. Facebook was initially created for students of Harvard College in order to keep in

touch, and then attracted a wider population and became even more popular than MySpace. The difference between MySpace and Facebook is that the latter focuses mainly on current and past real-life relationships brought on-line. Studies have identified that in average its members have a higher level of education than MySpace's. Maybe this is due to the fact that although the main functionality of these sites is similar, the actual experience is different: on Facebook people cannot decorate their own page and they have to add widget-like "applications" to add content on their page. Some applications can leverage the personal profile information and social links, bringing new opportunities to exchange and interact with our friends. For example, the "Movies" application allows one to rate movies; another applications allow friends to compare their movie tastes to evaluate their compatibility, or photos albums tagging (names, descriptions, ranking of friends in several domains, exchange of virtual gifts and small games). An interesting aspect of Facebook is the viral spreading of applications: people who add an application can invite their friends to join the same application so that they can interact with it. Every action is traced to one's public mini-feed, so that his friends can see what she/he is up to and what application she/he is using. The "call-to-action" consists of showing features (brought by an application) on users' profile page so that other users can join (and thus add the application) is an innovative incentive model. Applications can leverage user profile information from the Facebook platform; they can advertise and also charge users for premium features. A downside of this approach is that many applications abuse this model by forcing the user to invite friends before activating the expected feature. Hence most received applications invitations are not genuine and thus they lose impact, by being finally considered as spam. The number of publicly available applications grew exponentially, and there were almost 18,000 of them at the moment this article is being written. We can identify four major factors that can justify the success of MySpace and Facebook social networks:

- **Leverage the need of Social Ego Boost.**
- **Catch the user's Attention and Gather Profile Information** by providing fun content and applications. YouTube is a good example in this factor because watching videos becomes an addiction when surfing to proposed "relevant videos". On MySpace this is mostly done by users who decorated their profile page. On Facebook, the fun is brought by the applications.
- **Spread Messages in a Viral Fashion**, as people actions are visible to their friends through the mini-feed.
- **"Call-to-action"**, as people can interact with a friend's application before actually adding it. Adding an application must be straightforward: neither email address nor password to register, such information would have already been collected by the underlying platform (e.g. Facebook).

Overall, it seems that the winning guideline for a successful VC is that people have fun using it; they can keep in touch and interact with the people they know and connect with people they may like.

2.5 Monetizing the Community

VCs, as the rest of the Web 2.0 applications, suffer from a lack of good business model. For example, Facebook, with more than 120 million active users has projected earnings

before interest, taxes, depreciation and amortization (EBITDA) of $50 million for 2008 and a projected negative cash flow of $150 million for 2009. In the traditional economy, there are not many companies having such a large number of clients and such a critical financial situation. The lack of revenue is the biggest issue for VCs. Turning a community website into a money machine is not an easy task, therefore finding good revenue sources is a key issue. There are four main revenue sources for VCs:

- **Advertisement**
- **Subscription Fees**
- **Paid Features**
- **Selling User generated Content**

Advertising is currently the most common revenue sources, since it can be applied for almost every VC. Specialized communities have a higher profitability, since it's easier to target the consumer, advertiser are ready to pay higher CPM for those sites. Some other VCs require paying some fees at registration or monthly. Restricting features is also very common, for example users who want to increase their storage space on Flickr or Picasa have to pay a small monthly fee. Finally, for valuable user generated content, and if the VC is owner of the content, the content can be edited and sold in forms of books, videos, printings, t-shirts. An example is UrbanDictionary who recently published a book containing the best definition given by users. The terms of service leave the ownership to the submitter of the content but grant the company a non exclusive license to copy and sell the user generated content

3 Technical Dimension

3.1 Centralization vs. Decentralization

In current VCs the management model consists of a central entity taking care of all the low-level functionalities. However, the existence of a central server raises some significant issues related to privacy, censorship, and independence. Decentralized communities, based on the principles of peer-to-peer (P2P) systems, would attract users that value much these aspects. However, the fact that end-users are required to contribute a significant amount of resources for supporting core system functionality would be detrimental to the efficiency of the system and raises important incentive issues (e.g., free riding). We believe that web-based and self-organized communities should not be treated as substitutes to centralized communities but rather as their complement. Web-based communities are probably the only way to manage global scale online communities of millions of users, while self-organized communities would be a good alternative for more medium sized communities with a sufficient number of pre-existing trust relationships. One way to bootstrap decentralized virtual communities is to rely on existing social networks (such as Facebook and MySpace) in order to benefit both from the social ties developed between members of large virtual communities.

3.2 Incremental Deployment

Deployment of the infrastructure should be made according to traffic expectations. An early overinvestment in the hardware or in the software development can lead to

disastrous consequences. Even if the explosion of traffic is hard to forecast, indicators of internet presence, popularity in Technorati, and existing traffic analysis provide an interesting way for forecast. Concerning the software deployment, the first set of features should also not be excessive. Providing a bountiful set of features can lose the user in never-ending menus and pages. Before deploying or even implementing new features, it is important to get user's feedback. Recent announcement of the Beacon advertisement system in Facebook spawned a long controversy about the use of private data for advertising purpose.

3.3 Downtime, Availability, Performance

Interruption of service for a community is synonym of death. For most of the communities, users connect every day and expect a permanent availability of the service. Short interruptions (few hours) will disturb the users and spread an image of non trustworthiness, whereas longer interruptions will lead users to migrate to competitor services. Performance is also a key factor for successful communities; the amount of exchanged data can quickly become very large, especially in picture or video sharing. A slow response time is off-putting for new users and can become a reason for members to switch to another community.

3.4 Context Awareness

Most social communications today deal with contexts, people informs their friends what they are doing on their blog, share their pictures of last trips using emails, send SMS to ask friends where they are. But all these communications remain manual so far. Getting rid of these communications means and focusing on real conversations instead would be a step forward. Context awareness is an answer to the automation of these contextual rituals. The idea of "context awareness" is to sample every possible piece of context information, in order to infer the current situation of the user, such as location, current activity, surrounding people and devices. With the growing popularity of mobile phones with advanced capabilities such as Blue-tooth, broadband internet access, cameras and GPS receiver, the possibilities of leveraging "real world" context information increase. In the scope of VCs, context awareness enables the implementation of:

- A social radar which would visualize interesting information about surrounding peers. This information may include status information that can improve communication.
- A social network that automatically gives updates to friends about location, encounters, and activity, according to user's privacy preferences.
- A collaborative map on which users give some contextual information in exchange of useful services. As an example, a company could buy contextual information as an implicit way of gathering feedback and statistics on usage of their products and services, in order to improve them.
- A world of social recommendations that helps a person on the move decide where to eat or what movie to watch based on the recommendations of other community members. This is also a business opportunity for targeted advertising.

These opportunities can be met by inferring actual situations from sampled context data using the mobile phone (e.g. Bluetooth to discover surrounding phones and devices, GSM or WiFi positioning), the user profile, social graphs, and inference rules for reasoning pro-actively with all this knowledge. We think that context awareness is an opportunity to enrich VCs experience.

3.5 Integrating User Experience

The feedback of user experience within a VC can be gathered implicitly or explicitly. Indeed, on a platform with many applications like Facebook, the user experience with applications has significant impact on their spreading and active use. It is easy to gather implicit statistics due to the viral spreading of applications using invitations. Users can also provide explicit feedback by commenting, rating and reporting applications according to their expectations. With all this collaborative data, users can already evaluate the popularity, quality and usefulness of applications before actually adding them.

4 Social Dimension

4.1 Profiles Management

The usual way of taking part in a VC is to create a profile which stands as a personal avatar with a chosen name (a ”nickname” or ”pseudonym”) and possibly fictitious personal information. Entering multiple communities implies creating many profiles that are independent and not necessarily containing the same information. This approach is relevant when joining communities that are focused on specific domains, so that the user professional information will not be part of her/his leisure profile and vice versa. However, at the era of the social networks there is an emerging need to federate our identities. Indeed, this need is justified for many reasons:

1. The user has to remember her/his authentication credentials for every community. If he/she decides to use the same credentials on every platform, then one security breach will have considerable impact.
2. The change of a piece of information (e.g. current status, e-mail address or location) each profile of every community must be updated separately. Otherwise, inconsistencies between profiles will occur.
3. Communities usually provide internal messaging capabilities which are not interoperable. Keeping up to date with messages implies logging on all my communities.

Microsoft, Facebook and Google have been proposing their own unified account to federate one’s identity on various communities. Their unified accounts allow single-sign-on and consistent profile information but one account maps to one identity which is shared across communities as a same basic profile. This approach prevents the user from having separate identities, especially on Facebook where the user profile is richer than elsewhere. OpenID is an interesting alternative to identity federation because it is decentralized and not affiliated with any big player of the IT industry, and thus seen as ”not evil” concerning the usage of your personal data. A ”persona” (identity) can be hosted on any OpenID container website, and this website will be used to authenticate

the user on any third party OpenID-compliant website. The identify management is thus delegated to the OpenID container instead of the community website itself.

4.2 Privacy and Anonymity

The part of the user activity revealed to the interested parties and/or made public (e.g., visits, when a user is online, profile information, etc.) could affect the way people behave in VCs. Increased visibility strengthens the personal responsibility and the opportunities for social interactions. However, increased transparency raises privacy issues; besides, private information (identity and/or content) is being stored in central databases which could be exploited for commercial purposes, or could be exposed through security breaches [10]. A VC should take care of privacy concerns seriously and be transparent about its privacy policies during the subscription; besides, as VC can leave for members to decide what is private and what is not. Some third-party applications, such as in Facebook, lead to malicious data harvesting, current protocol often forces users to give application access to non required data [11].This is a very crucial issue; for instance, Facebook having more than 110 million active users, a popular application developed by a malicious company can gather huge amount of private data.[11] proposed a simple privacy-by-proxy approach to help keeping privacy while providing required information to third-party applications.

Notice also that the professional networking website LinkedIn keeps profiles anonymous until the user is recognized as a part of the social graph.

4.3 Acceptance

One of the key questions in VC research is why some systems are accepted and some are rejected by users. The factors and processes affecting to users adoption and use, have received a lot of interest form IT researchers. Scholars have developed several general acceptance models which link individual reactions and intentions to actual use of the system [12]. Probably the most popular acceptance framework is Technology Acceptance Model (TAM) [13]. According to TAM perceived Ease Of Use (EOU) and usefulness are the sole determinants of attitudes towards an innovation, which in turn predicts the behavioral intention that is a solid predictor of actual behaviour. Information systems researchers have developed many extensions to the original TAM, and new intention determinants to the original model were added to cover the special features of the analyzed context. Examples of extensions include perceived credibility [](Wang et al. 2003), trust [14], playfulness [15,16], self-expresiveness [17] and enjoyment [18]. This rises a question what are the special characteristics of virtual communities that affect their success and acceptance?

The factors resulting either in the success or failure of VCs are still unclear. However, one of the critical factors determining the success of a VC is its members' active information sharing and generation [19,20,21,22]. SSuccessful applications have many active users. Based on discussion above we have identified three intention determinants to our extended TAM model: perceived value, perceived ease of use and perceived social enjoyment. Although we base our discussion on TAM we follow the logic of Kaasinen [23] and replace the original determinant of TAM (perceived usefulness) with perceived value.

This way we emphasize that instead of implementing a collection of useful features, the designers of VCs should focus on key values provided to the members of the community.

Perceived Value. The value of a VC is a critical aspect for attracting users to be active participants in the VC. New communities should offer a clear added-value to its potential members. In a VC value is generated from the software itself and from the members' contributions (in terms of content, expertise, and presence). Based on the work of [24] we can identify two categories of value: pragmatic and hedonic value. Pragmatic value refers to the VC's usefulness and it includes practical value aspects (e.g. ablility to share information or generate knowledge from the interactions with other members). Hedonic value, on the other hand, addresses human needs for excitement (novelty, change) and pride (social power, status). Pragmatic value refers to the VC's usefulness and it includes practical value aspects (e.g. ability to share information or generate knowledge from the interactions with other members). Practical value could be independent of the existence of other users or dependent on them. To take the example of Flickr, users creating an account in Flickr have immediately the ability to backup their photos and show them to their friends and family, independently of how many more users are members of Flickr. Similarly, Delicious offers the possibility to users to view their bookmarks from any computer connected to the Internet. This type of value is important for the bootstrapping of the system. On the other hand, knowledge, feedback, expertise generated from the interactions with other members or just by "lurking" is another significant value generator of a VC. For example, Flickr offers to members the opportunity to learn photography, improve heir own skills. The opportunities for socialization (even if not the primary objective of the community) and self improvement are strong incentives for users to participate. Hedonic value, on the other hand, addresses human needs for excitement (novelty, change) and pride (social power, status). In VC context the importance of hedonic value is more important than in information systems in general. VCs are not expected to serve only the members' needs for communication and information but also for socialization, emotional connections, entertainment, fun, and pride [25].

Perceived Ease of Use. Perceived ease of use is the second intention determinant of our model. There has been a rich stream of EOU studies in all kinds of information systems during the last decades; their main goal has been to create products that have high usability. Usability increases customer satisfaction and productivity, leads to customer trust and loyalty, and contributes to tangible cost savings and profitability. Thus, a high usability can also lead to the success of a VC. Because information sharing is essential to all VCs, a successful VC must offer easy to use communication tools which help people to understand each other in the on-line community. Today's most popular communication methods in VCs are still text-based although other forms of communication, such as voice or video conferencing can also be used. Several studies have examined the promotion of mutual understanding in text based communication [26,27,28,29]. The method for promoting mutual understanding can be categorized into two: the enhancement of the text presentation (e.g. adding visual attributes to text such as changing size or color of fonts) and the design of statement database (e.g. add explicit statements or symbols to database like the "smiley"). In successful VCs the interaction between the user and the system must be entertaining, engaging and effective experience.

Perceived Social Enjoyment. The above discussion revealed that in VC both value and ease of use have a strong social dimension. The perceived value of a VC is not limited only to personal values independent of the existence of other users but in includes values related to the community, and to common outcomes of the community that offers satisfaction and pride for the members that took part in its production; such as Wikipedia top contributor, and Flickr's explore page. Clay Shirky calls this the "promise" of the community [30]; the way this "promise" is expressed and communicated can play an important role in the users' participation. Similarly ease of use is not limited to simple ability to the use the system; indeed, in successful VCs the interaction between the user and the system offers an entertaining, engaging and effective experience. We believe that VCs must support sociability and enjoyment throughout the activity, and that the perceived social enjoyment is an essential intention determinant and prerequisite of actual use of the system. Although some studies on methods to measure social enjoyment exists [31] further studies in this context are still needed.

5 Conclusions

We discussed in this paper three main characteristics to enable successful VC projects. For the business part, we stressed the importance of being a first mover on the market or to propose a clear and important added-value in order to be competitive. Good relationships with influent bloggers and specialized web media are keys to good visibility. We also discussed strategies on how to attract users, as well as on the financial dimension. On the technical side, we had few suggestions regarding the design and development and the traffic forecast. Furthermore, we suggested that context-awareness is an opportunity to develop exclusive and powerful added value VCs that can be competitive; besides, we emphasized the importance of users' feedback during the process of development of new features. Moreover, we approached the social dimension from the user's perspective; thus, we've discussed several identification mechanisms that allow users to manage their identity/profiles; then we've overviewed trust, privacy and anonymity needs; finally, we presented an acceptance model that may give insight into what makes a VC successful from a user point of view.

References

1. Preece, J.: Online Communities - Designing Usability, Supporting Sociability. John Wiley & Sons, Ltd., Chichester (2000)
2. Weissman, D.: A social ontology. Yale University Press, New Haven (2000)
3. ElMorr, C., Kawash, J.: Mobile virtual communities research; a synthesis of current trends and a look at future perspectives. Int. J. Web Based Communities 3, 386–403 (2007)
4. Kaji, N., Ragab, K., Ono, T., Mori, K.: Autonomous synchronization technology for achieving real time property in service oriented community system. In: The 2nd International Workshop on Autonomous Decentralized System, 2002, pp. 16–21 (2002)
5. Sousa, J.P., Garlan, D.: Aura: an architectural framework for user mobility in ubiquitous computing environments. In: WICSA, pp. 29–43 (2002)
6. Li, Y., Leung, V.C.M.: Supporting personal mobility for nomadic computing over the internet. SIGMOBILE Mob. Comput. Commun. Rev. 1, 22–31 (1997)

7. Keranen, H., Rantakokko, T., Mantyjarvi, J.: Sharing and presenting multimedia and context information within online communities using mobile terminals. In: Proceedings of 2003 International Conference on Multimedia and Expo., 2003. ICME 2003, vol. 2, pp. II- 641–II-644 (2003)
8. Cole, H., Stanton, D.: Designing mobile technologies to support co-present collaboration. Personal Ubiquitous Comput. 7, 365–371 (2003)
9. Abdul-Rahman, A., Hailes, S.: Supporting trust in virtual communities. In: Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, 2000, vol. 1, p. 9 (2000)
10. Rosenblum, D.: What anyone can know: The privacy risks of social networking sites. IEEE Security & Privacy 5, 40–49 (2007)
11. Felt, A., Evans, D.: Privacy protection for social networking platforms. In: Web 2.0 Security and Privacy 2008, Oakland, CA (2008)
12. Venkatesh, V., Morris, M.G., Davis, G.B., Davis, F.D.: User acceptance of information technology: Toward a unified view. MIS Quarterly 27, 425–478 (2003)
13. Davis, F.D.: Perceived usefulness, perceived ease of use, and user acceptance of information technology. MIS Quarterly 13, 319–340 (1989)
14. Dennis, C.E., Alsajjan, B.A.: The impact of trust on acceptance of online banking. In: European Association of Education and Research in Commercial Distribution, Brunel University (2006)
15. Moon, J.W., Kim, Y.G.: Extending the tam for a world-wide-web context. Information & Management 38, 217–230 (2001)
16. Cheong, J.H., Park, M.C.: Mobile internet acceptance in korea. Internet Research: Electronic Networking Applications and Policy 15, 125–140 (2005)
17. Pedersen, P., Nysveen, H.: Usefulness and self-expressiveness: Extending tam to explain the adoption of a mobile parking service. In: Proc. of 16th Bled Conference, Bled, Slovenia (2003)
18. Phuangthong, D., Malisawan, S.: A study of behavioral intention for 3g mobile internet technology: Preliminary research on mobile learning. In: of the 2nd International Conference on eLearning for Knowledge-Based Society, Bangkok, Thailand (2005)
19. Ardichvili, A., Page, V., Wentling, T.: Motivation and barriers to participation in virtual knowledge-sharing communities of practice. Journal of Knowledge Management 7, 64–77 (2003)
20. Bross, J., Sack, H.: Encouraging participation in virtual communities: The"it-summit-blog"case. In: Proc. of IADIS Int. Conf. e-Society 2007 (2007)
21. Moore, T.D., Serva, M.A.: Understanding member motivation for contributing to different types of virtual communities: a proposed framework. In: CPR, pp. 153–158 (2007)
22. Koh, J., Kim, Y.G., Butler, B., Bock, G.W.: Encouraging participation in virtual communities. Commun. ACM 50, 68–73 (2007)
23. Kaasinen, E.: User acceptance of mobile services - value, ease of use, trust and ease of adoption. VTT Information Technology (2005)
24. Hassenzahl, M., Kekez, R., Burmester, M.: The importance of a software's pragmatic quality depends on usage modes. In: Proc. of the 6th International Conference on Work with Display Units, Berchtesgaden Germany (2002)
25. Antoniadis, P., Grand, B.L.: Incentives for resource sharing in self-organized communities: From economics to social psychology. In: ICDIM, pp. 756–761 (2007)
26. Farnham, S., Chesley, H.R., McGhee, D.E., Kawal, R., Landau, J.: Structured online interactions: improving the decision-making of small discussion groups. In: CSCW, pp. 299–308 (2000)

27. Vronay, D., Smith, M.A., Drucker, S.M.: Alternative interfaces for chat. In: ACM Symposium on User Interface Software and Technology, pp. 19–26 (1999)
28. Toth, J.: The effects of interactive graphics and text on social influence in computer-mediated small groups. In: Proceedings of ACM CSCW 1994 Conference, pp. 299–310 (1994)
29. DiMicco, J.M., Lakshmipathy, V., Fiore, A.T.: Conductive chat: Instant messaging with a skin conductivity channel. In: Extended Abstracts of ACM CSCW 2002 Conference, pp. 193–194 (2002)
30. Shirky, C.: Here Comes Everybody: The Power of Organizing Without Organizations. The Penguin Press HC (2008)
31. Lindley, S.E., Monk, A.F.: Social enjoyment with electronic photograph displays: Awareness and control. International Journal of Human-Computer Studies 66, 587–604 (2008)

A Multiagent-System for Automated Resource Allocation in the IT Infrastructure of a Medium-Sized Internet Service Provider

Michael Schwind[1] and Marc Goederich[2]

[1] BISOR, Technical University Kaiserslautern
Erwin-Schrödinger-Str. Geb. 42, D-67663 Kaiserslautern, Germany
schwind@wiwi.uni-kl.de
http://www.bisor.de
[2] Root eSolutions s.à.r.l, Rue John F. Kennedy 35, L-7327 Steinsel, Luxembourg
mg@root.lu
http://www.root.lu

Abstract. In this article we present an agent-based system that is designed for the automated allocation of web hosting services to the IT resources at a medium-sized Internet service provider (ISP). The system is capable of finding a cost minimizing allocation of web hosting services on the distributed IT infrastructure of the ISP. For this purpose an agent which can independently determine a price for each package of web hosting services is assigned to each resource. The allocation mechanism employs a system of price and cost functions to form an economic model which guarantees a continuous capacity load for the companies' IT resources. According to the demand for web hosting services, resource agents can invest in the acquisition of IT infrastructure. These investments have to be amortized by the resource agents using the returns yielded by the web services sold to the ISP customers. By using real world demand profiles for web services packages taken from the operational systems of a medium-sized ISP, we were able to prove the stability of the resource allocation system.

Keywords: Multiagent-System, Automated Resource Allocation, Web-Services, Internet Service Provider.

1 Introduction

The widespread use of broadband Internet access points in private households led to an increasing demand for web hosting services because more and more consumers establish their individual web presence. Internet service providers (ISP) reacted to the added demand for web service capacity by offering web services where multiple customers share virtually separated web services based on the joint IT infrastructure of the ISP. This service model is already very popular for ISP-based service provision by small and medium sized companies [8]. Nevertheless, even the use of shared web services leads to a strongly growing demand for IT infrastructure at the service providers. In this context it is a crucial problem for the ISPs to exactly determine the demand

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 689–703, 2009.
© Springer-Verlag Berlin Heidelberg 2009

for investments in specific IT infrastructure components and to allocate the additional resources to the services provided. For this reason we present an auction-based system that has been designed for the automated allocation of web hosting services to the IT resources of the medium-sized ISP Root eSolutions (www.root.lu). The system is capable of allocating the web hosting services on a distributed IT infrastructure in such a way that the load-dependent cost for the resources use is minimal. In order to do this, software agents are assigned to the resources in the ISP's IT infrastructure. The agent-based mechanism employs a system of price and cost functions forming an economic model that guarantees a continuous capacity load distribution of the IT resources. Following the exogenous demand for web hosting services, these agents can also invest in new IT infrastructure if necessary. These investments have to be amortized by the resource agents using the returns yielded from the web services sold to the ISP customers. By using real world demand profiles taken from the operational systems of a medium-sized ISP, we are able to prove stability of the allocation system.

2 Automated Economic Resource Allocation

With the evolution of large computer networks at the beginning of the 80^s the first work on economic-based systems for the allocation of jobs on distributed IT resources emerged. The seminal article 'Incentive Engineering for Computational Resource Management' by [4] and the book 'Market-based Control' is a crucial basis for this research area. [12] distinguishes between two main categories of economic resource allocation in distributed computer systems: allocation games and market-based allocation. In allocation games the participants are competing for direct access to scarce resources, while payments for the outcome of the game are the crucial optimization objective of the agents' strategies, whereas in market-based allocation processes the market price of the use of the resources is the decisive variable for the allocation process. Well known examples for the group of allocation games are Nash Games in networks [1] and the application of the iterative prisoner dilemma for resource allocation in a peer-to-peer network [8]. For the market-based allocation process three main approaches can be identified: multilateral negotiations, tâtonnement processes and auctions. The ContractNet protocol is often used for a resource allocation process that is managed using multilateral negotiation between agents in a distributed computer system [3]. [5] present an example of such a communication intensive negotiation approach. An example for a tâtonnement-based allocation process is the WALRAS mechanism designed by [2]. Their approach uses a centralized market maker that iteratively equilibrates supply and demand of different resources in an auction-like process. Auction protocols are the most common type of protocol for automated resource allocation in distributed IT systems. Simple implementations use forward, reverse and double auctions including English, Dutch and Vickery protocol types [7,13,11,10]. More sophisticated auction protocols take the complementarities between the different resource types into consideration. Such complex allocation mechanisms are normally based on the application of combinatorial auctions [14].

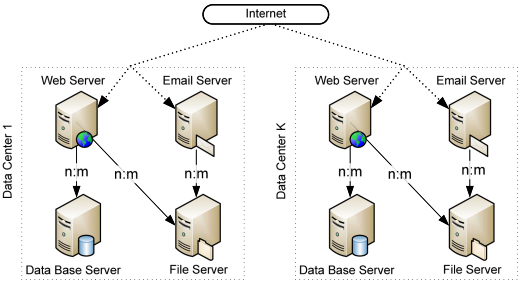


Fig. 1. IT system architecture used at 'Root eSolutions'

Table 1. Web hosting packages offered by 'Root eSolutions'

Web hosting package	**small**	**medium**	**large**
Hard disk memory (MB)	100	500	1000
No. e-mail accounts	10	50	100
No. data bases	2	4	6
Leasing fee per year	€50	€100	€150

3 Web Hosting System for a Medium-Sized ISP

For more than a decade web hosting[1] has been a common IT service, renting data space, server resources, and broadband network capacity to Internet users. Normally, these resources are sold in the form of different types of bundles and package sizes including several additional services, like Email accounts, content management systems, or databases. In analogy to the highly differentiated range of services packages, the price structure of web services offered by the ISPs is extremely heterogeneous. Fig. 1 shows an example of the IT system architecture used at the medium-sized ISP *'Root eSolutions'*. The ISP infrastructure consists of several autonomous data centers that work in parallel: The core of such a data center is a high capacity file server connected to an e-mail and a web server via a high speed network. Additionally, the web server has access to a MySQL data base server. Root eSolutions sells different combinations of these services in the form of web hosting packages. Tab. 1 shows three basic packages currently offered to Root eSolutions clients together with their capacity specifications and the leasing fee associated.

4 Agent-Based Resource Allocation Model

4.1 Multi-agent Model

In this section we describe the agent-based model which is designed to allocate web hosting packages on the distributed IT infrastructure resources of our medium sized ISP. As a first step we define the different types of agents that are responsible for the

[1] See: http://en.wikipedia.org/wiki/Web_hosting

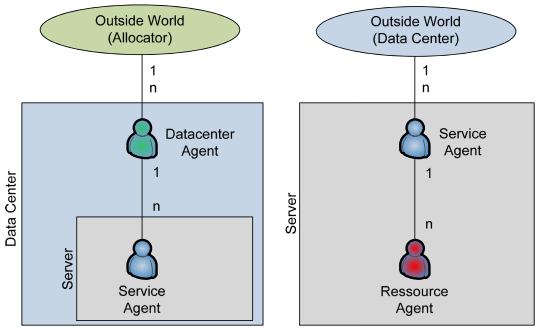


Fig. 2. Agents' environment: Visual field of the DCagent (left) and service agent (right)

allocation process. As a second step we present the corresponding goal, cost, and price functions which should guarantee that the auction process will end in the desired state of the allocation process. Regarding the system architecture depicted in Fig. 1, it is easy to determine an appropriate structure for the multi-agent system: Each server is endowed with a service agent which is responsible for allocating the required resource bundles to the incoming web service requests. For this purpose the service agent can communicate with the outside world (data center) and is also connected to other local resource agents in its server world (see Fig. 2). The service agent, which can only observe the behavior of agents in its own server world, collects *priceRequests* from outside and responds with a *returnPrice* for the resource package requested. The service agent itself is allowed to send a priceRequest to the other service agents, except for the type of service offered by itself. In order to determine the price for the resource offer, the service agent sends a query to the resource for which it is responsible (e.g. hard disk and processor capacity). The resource returns a status message about current utilization. Then the service agent states the bid price using a resource load coefficient calculated on the status information of the resource type managed by the server.

Fig. 3 shows typical hourly and monthly load profiles for the resources 'hard disk' and 'network capacity'. The hard disk load is practically constant on an hourly basis and can be considered as a continuously growing function from an annual perspective. Disk capacity is a critical resource in the allocation model because a full disk blocks the execution of the other services. The disk load is characterized by:

$$load = \frac{current{-}disk{-}use}{max{-}disk{-}capacity} \text{ or } load = \frac{prenotified{-}disk{-}use}{max{-}disk{-}capacity}.$$

The situation is different for the use of the network capacity of a web hosting package. There is normally a high level of fluctuation in the use of network capacity, with a peak load in the middle of the day and in the evening. The typical long term demand profile for network capacity use is similar to the lifetime cycle of a product. After some months of a strong increase in network capacity use, the demand curve truncates and remains on a high level. Network capacity use is a critical factor only at peak load times. In the worst case there are only long latency times for the user of the web hosting service. In addition to the agents responsible for the management of the four service types

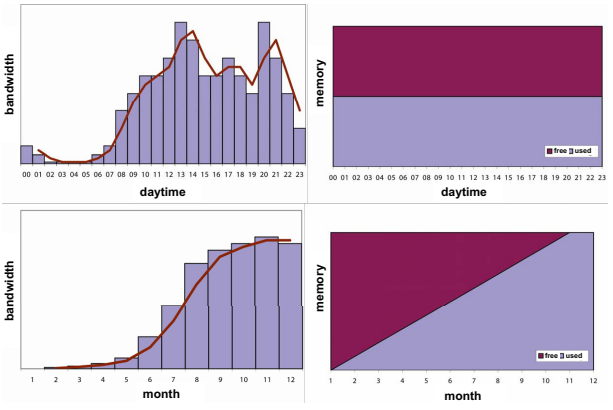


Fig. 3. Hourly (up) and monthly (low) load profiles for hard disk (right) and network capacity (left)

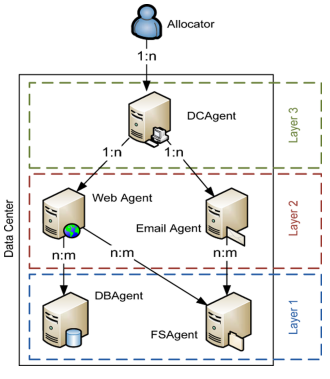


Fig. 4. Hierarchically layered structure of the agent-based resource allocation system

like the file server (FSAgent), the data base (DBAgent), the web service (WebAgent), and the email server (EmailAgent) we introduce the data center agent (DCAgent) and the allocator agent (allocator). The DCAgent's view goes beyond the walls of its own data center.

The DCAgent collects the priceRequests from the allocator and forwards them to the WebAgent and the EmailAgent. After having received messages with the resource prices from the service agents, the DCAgent selects the lowest offers for email and the web service resources. The DCAgent calculates the resulting offer price by adding network bandwidth costs. Finally the DCAgent submits three bids to the allocator: a bid price for the single email service package, a bid price for a single web service package and a bid price for a combined email and web service package. DCAgents are also responsible for decisions concerning investments in new IT infrastructure. If the average server load is at the upper capacity limit, new infrastructure components have to be acquired by the DCAgent.

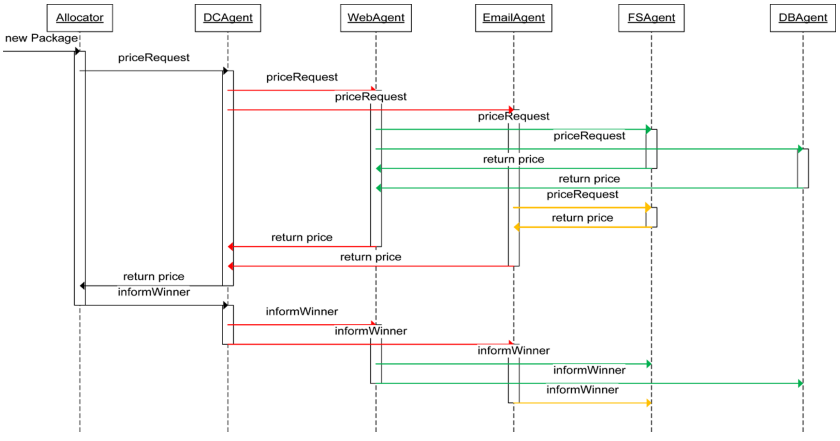


Fig. 5. UML diagram of the message flow in the agent-based resource allocation system

The task of the allocator agent is to dispatch the resource demand to the different data centers in the ISP's infrastructure so that the costs of the resources become minimal. In order to do this, the allocator agent is positioned outside the data centers and sends priceRequests to the DCAgents. See Fig. 4 for an illustration of the hierarchical structure of the resource allocation system. After having received the offers from the DCAgents the allocator selects the cost minimal combination for the resource packages. This combination may include resources from several data centers. The allocator is obliged to procure the resource capacities that are necessary to provide the web hosting services requested by the ISP's clients even if the resource costs exceed the return yielded from the sale of the web hosting contracts. The only way for the allocator to reduce resource costs in such a situation is to invest in IT capacities by opening a new data center in order to reduce the average resource load.

Fig. 5 summarizes the message flow in the agent-based resource allocation system using an UML sequence diagram. The resource requests cascade down the hierarchical structure of the allocation system. When the auction has taken place the service agents with the winning bids are informed in order to allow them the opportunity to reserve the promised resource capacities.

4.2 Formal Model Description

Each of the six agent types described in the previous section uses a particular cost function to calculate the bid price in the allocation process. We differentiate between nonrecurring costs occurring only once in the lifetime of a server system and periodic cost that is counted on an annual base. Expenditures for the initial acquisition of hardware are nonrecurring costs. However, at the end of its technical life-span (up to three or five years) the server system must be replaced, making a new investment necessary. Installation costs are included in investment expenses, whereas maintenance expenditures can be considered as periodical infrastructure costs. Energy expenses (electrical power and air conditioning), expenditure for infrastructure (rental of capacity) and internal

Table 2. Cost categories for different agent types

Costs	DBAgent	FSAgent	EMAgent	WebAgent	DCAgent
Invest.	AK_{DB}	AK_{FS}	AK_{MAIL}	AK_{WEB}	AK_{DC}
Energy	EK_{DB}	EK_{FS}	EK_{MAIL}	EK_{WEB}	EK_{DC}
Infrastr.	IK_{DB}	IK_{FS}	IK_{MAIL}	IK_{WEB}	IK_{DC}
Network	SK_{DB}	SK_{FS}	SK_{MAIL}	SK_{WEB}	SK_{DC}

network capacity are periodic costs. Tab. 2 shows the variable names for the different categories of cost that will be used in following sections to describe the behavior of the different agent types in the allocation system.

DBAgent. The goal of the DBAgent is to maximize the return yielded by the acceptance of resource bids by the web agent. This return coincides with periodic costs for investments, infrastructure and internal network capacity as well as investments. A higher bid price leads to an increased contribution margin on the one hand, but reduces the probability of bid acceptance on the other hand. Let max_{DB} be the maximum number of database services that can be hosted on a server. In the case of a full server load in period j_{DB} the total cost of the database would add up to the yearly amount of:

$$k_{DB} = \frac{\frac{AK_{DB}}{j_{DB}} + EK_{DB} + IK_{DB} + SK_{DB}}{max_{DB}}$$

The load of a database server at the beginning of the allocation process is set to zero. The DBAgent has no estimator for future workload until it has gathered some experience of the arrival rate of the incoming resource requests and the server size required by the database packages. The DBAgent starts with an estimator for the expected workload a_{DB} based on the ISP's experience. The estimator will be adjusted to more appropriate values during the system's operation phase. The load of a server is defined as $a_{DB} = \alpha_{DB}/max_{DB}$, where α_{DB} is the number of database services currently hosted on the server. In order to reflect the bottleneck character of the data base we use an exponential function to calculate the bid prices for resource requests at the DBAgent:

$$exp_{DB} = e^{(e_{DB} * a_{DB})}$$

The exponent e_{DB} of the cost multiplier function exp_{DB} depends on the importance of the database for the functioning of the ISP infrastructure. The parameterization of the cost multiplier function is exogenously given by the system designer and varies for different types of service agents, reflecting the criticality of a resource in the case of its being in short supply in the allocation system. A side effect of the construction of the cost multiplier function chosen in our model is that the fast rising bid price near the capacity limit of a resource type helps to gather cash for further investments in the ISP's infrastructure in order to create additional resource capacities. The pricing function of the service agents also includes a discount factor for larger resource packages. In the case of the DBAgent, the discount factor is set to $r_{DB} = f(x_{DB})$ $(0 \leq r_{DB} \leq 1)$ where x_{DB} is the number of databases offered in a web hosting package. The discount factor reflects the customer price structure for web hosting services given in Tab. 1. A

further factor for the calculation of the bid price of the DBAgent is the profit margin $g_{DB} \geq 1$. The profit margin is subject to a dynamic adaptation process: If an agent receives consecutive acceptance of its bids, it seems to be obvious that other servers have a higher capacity load and it can raise its profit margin without losing subsequent bids. The agents' strategy is to raise the profit margin as long as it wins the bids and to lower it if it loses bids in the allocation process. The formula used by the DBAgent to calculate a bid price for a resource request is finally written as:

$$p_{DB} = k_{DB} * exp_{DB} * r_{DB}(x_{DB}) * g_{DB} * x_{DB}$$

FSAgent. The mechanism of the FSAgent is constructed in analogy to the DBAgent including the profit maximization goal. max_{FS} stands for the disk storage capacity available (in mega byte) and x_{FS} is the amount of disk storage capacity in a web hosting package requested. The price for a megabyte of disk storage at high level load of the server is:

$$k_{FS} = \frac{\frac{AK_{FS}}{j_{FS}} + EK_{FS} + IK_{FS} + SK_{FS}}{max_{FS}}$$

The current capacity load ratio $a_{FS} = \alpha_{FS}/max_{FS}$ is included in the cost multiplier while α_{FS} represents the disk capacity already in use:

$$exp_{FS} = e^{(e_{FS} * a_{FS})}$$

After defining the profit margin g_{FS} and the discount rate $r_{FS} = f(x_{FS})$ $(0 < r_{FS} \leq 1)$ we can calculate the bid price for a resource request at the file server:

$$p_{FS} = k_{FS} * exp_{FS} * r_{FS}(x_{FS}) * g_{FS} * x_{FS}$$

EmailAgent. The EmailAgent is a profit maximizer and belongs to the group of agents that have to acquire external resources in order to offer their services. The EmailAgent requires additional file server capacity. The maximum number of accounts hosted on an email server is given by max_{MAIL}. The value of max_{MAIL} is subject to an adaptation process in analogy to the value of max_{DB}. The cost calculation follows the FSAgent:

$$k_{MAIL} = \frac{\frac{AK_{MAIL}}{j_{MAIL}} + EK_{MAIL} + IK_{MAIL} + SK_{MAIL}}{max_{MAIL}}$$

The capacity load $a_{MAIL} = \alpha_{MAIL}/max_{MAIL}$, the multiplier $exp_{MAIL} = e^{(e_{MAIL} * a_{MAIL})}$ and the discount function $r_{MAIL} = f(x_{MAIL})(0 < r_{MAIL} \leq 1)$, as well as the profit margin g_{MAIL}, are defined in analogy to the FSAgent.

In order to procure the file server capacity which is necessary to provide the memory space for the email accounts requested by the DCAgent, the EmailAgent selects the bid with the minimal price from the offers available, after having sent a priceRequest to the of the FSAgents:

$$k_{MAIL,FS} = min(p_{FS_1}, p_{FS_2}, ..., p_{FS_n})$$

The EmailAgent includes the expenses for the procurement of the file server capacity in its bid price calculation:

$$p_{MAIL} = (k_{MAIL,FS} + k_{MAIL} * exp_{MAIL} \quad (1)$$

$$*r_{MAIL}(x_{MAIL}) * x_{MAIL}) * g_{MAIL} \quad (2)$$

x_{MAIL} is the number of email accounts requested in the bid for resources.

WebAgent. It is responsible for the procurement of the resources necessary to provide the web services requested by the DCAgents. While fulfilling this function, the WebAgent sends priceRequests to FSAgents and the DBAgents. In the first step the WebAgent calculates the server costs per web site:

$$k_{WEB} = \frac{\frac{AK_{WEB}}{j_{WEB}} + EK_{WEB} + IK_{WEB} + SK_{WEB}}{max_{WEB}}$$

In this formula g_{MAIL} is the maximum number of web sites that can be hosted on the server administered by the WebAgent. The WebAgent calculates costs using the server load $a_{WEB} = \alpha_{WEB}/max_{WEB}$, the multiplier $exp_{WEB} = e^{(e_{WEB} * a_{WEB})}$ and the discount rate $r_{WEB} = f(x_{WEB})(0 < r_{WEB} \leq 1)$.

Like the EmailAgent the WebAgent has to buy resource capacities from the FSAgents, generating the following costs: $k_{WEB,FS} = min(p_{FS_1}, p_{FS_2}, ..., p_{FS_n})$. It has to procure data base resources: $k_{WEB,DB} = min(p_{DB_1}, p_{DB_2}, ..., p_{DB_n})$. The price function of the WebAgent includes a profit margin g_{WEB}:

$$p_{WEB} = (k_{WEB,FS} + k_{WEB,DB} + k_{WEB} *$$
$$exp_{WEB} * r_{WEB}(x_{WEB}) * x_{WEB}) * g_{WEB}$$

DCAgent. The DCAgent is the most complex agent in our allocation system. As the highest level of the data center hierarchy profit maximization is its major goal. The DCAgent acts as a consumer of email and web server resources and is also responsible for the administration of the bandwidth capacity. The allocation of bandwidth is carried out stepwise. The bandwidth pricing model is exogenously given by the tariff model of the telecommunication provider. The bandwidth fee consists of a fixed charge for the line capacity and variable costs for data traffic. A higher bandwidth capacity brings lower variable costs for data traffic and higher charges for line capacity and vice versa. The DCAgent has a cost structure that is constructed similar to that of the other agents:

$$k_{DC} = \frac{\frac{AK_{DC}}{j_{DC}} + Ek_{DC} + Ik_{DC} + Sk_{DC}}{max_{DC}}$$

max_{DC} is defined as the maximum bandwidth to the external world currently available for the ISPs infrastructure. The consumption of bandwidth resources required for a web hosting package depends on the number of email accounts and the hard disk memory size included in the package. In the following discussion we introduce the variable β_{MAIL} to describe the annual bandwidth consumption per email and β_{WEB} to indicate the annual bandwidth consumption per megabyte hard disk memory size in the web

hosting package. The DCAgent estimates the values of these variables at the beginning of the allocation process and adapts them to empirically retrieved values later on. The data center agent uses an exponential cost factor $exp_{DC} = e^{(e_{DC} * a_{DC})}$ with $a_{DC} = \alpha_{DC}/max_{DC}$ and a function analogous to the other agents' cost function cost factor $r_{DC} = f(x_{DC})(0 < r_{DC} \leq 1)$. The internal discount function is logical for the calculation of bandwidth capacity costs because it reflects the tariff structure of the external provider of bandwidth capacity. The profit margin of the DCAgent is g_{DC}. The DCAgent's formulas for the cost calculation of the email and web resources are:

$$k_{DC,MAIL} = \min(p_{MAIL_1}, p_{MAIL_2}, ..., p_{MAIL_n})$$
$$k_{DC,WEB} = \min(p_{WEB_1}, p_{WEB_2}, ..., p_{WEB_n})$$

The DCAgent submits three bids types in the allocation process: a price for the email package, a price for the web package and a price for the combination of both resources. The price for the email package is:

$$p_{DC,MAIL} = (k_{DC,MAIL} + k_{DC} * \beta_{MAIL} * x_{MAIL} * exp_{DC} * r_{DC}(\beta_{MAIL} * x_{MAIL})) * g_{DC}$$

The web package is offered for the following price:

$$p_{DC,WEB} = (k_{DC,WEB} + k_{DC} * \beta_{WEB} * x_{WEB} * exp_{DC} * r_{DC}(\beta_{WEB} * x_{WEB})) * g_{DC}$$

The combination of both resource types is offered with a discount compared to the prices of the individual resource packages because of the decreasing cost for higher bandwidth capacities:

$$p_{DC} = (k_{DC,MAIL} + k_{DC,WEB} + k_{DC} * (\beta_{MAIL} * x_{MAIL} + \beta_{WEB} * x_{WEB}) * exp_{DC} * r_{DC}(\beta_{MAIL} * x_{MAIL} + \beta_{WEB} * x_{WEB})) * g_{DC}$$

The DCAgent is responsible for investments into new IT infrastructure that is required in the data center.

Allocator. The Allocator is bound to accept all incoming customer requests for web hosting services. Because it must accept the orders for a price p following the fixed structure scheme given in Tab. 1 the only possible way for the Allocator maximize its profit $p - k$ is to reduce the production costs for the web hosting packages. In order to do this, the Allocator selects the cost minimal resource package combination from the DCAgents' bids in analogy to a reverse auction.

$$k = min(min(p_{DC_1,MAIL}, p_{DC_2,MAIL}, ..., p_{DC_n,MAIL}) + min(p_{DC_1,WEB}, p_{DC_2,WEB}, ..., p_{DC_n,WEB})), min(p_{DC_1}, p_{DC_2}, ..., p_{DC_n}))$$

$p_{DC,EMAIL}$ is the bid price of a resource package on an email server and $p_{DC,WEB}$ is the bid price for a resource package on a web server, p_{DC} is the bid price for a

package that combines both resources. In the case of a negative profit, the Allocator is able to wait until market forces reduce the bid prices offered by the DCAgents or to invest in a new data center in order to reduce market prices.

5 Simulation Experiments

This section presents the results of simulation studies performed with the VISUAL C# implementation of our multi-agent allocation system. The experiments make use of real

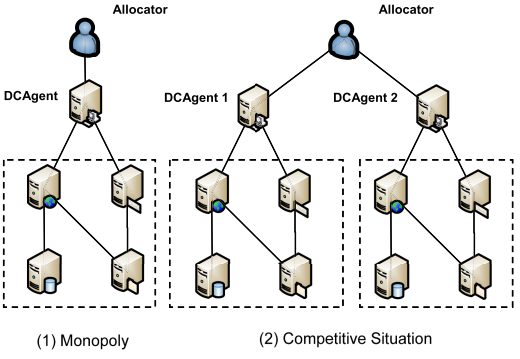


Fig. 6. System configurations for different types of economic situations of the DCAgent in the ISP's infrastructure: A monopoly (left) and a competitive situation (right)

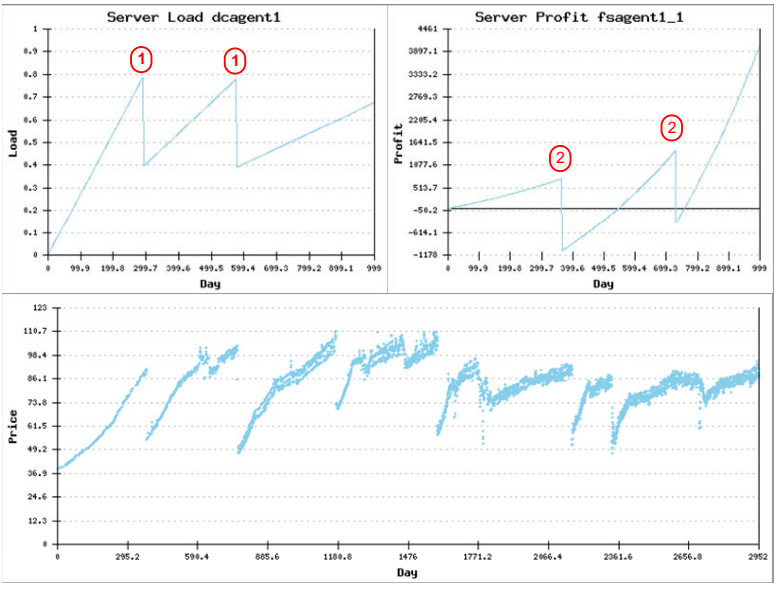


Fig. 7. Diagrams showing the server load of the DCAgent, the profit of the FSAgent, and the price for a medium size web hosting package in the course of time

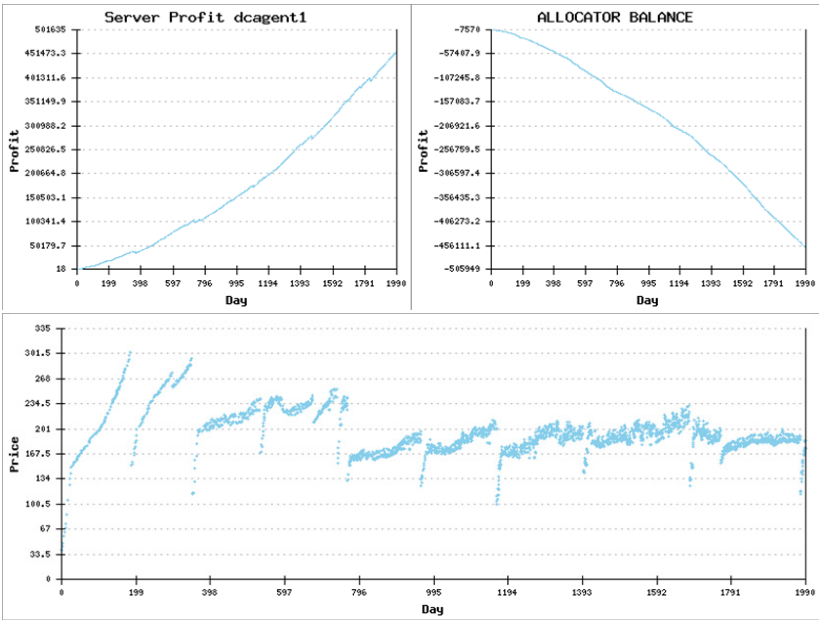


Fig. 8. Diagrams showing disequilibrium between the profit of the DCAgent and the Allocator in the monopoly situation (upper part) and the resulting price for a medium size web hosting package in the course of time (lower part)

world data representing the customer requests for web hosting services. In order to test the functional capability of the allocation system we use two configurations shown in Fig. 6: The first configuration (1), with a single data center reflects a *monopoly*, while the second configuration (2) with two data centers represents a *competitive system*.

The simulation experiments are conducted as follows: Each customer contract for a web hosting service expires after 365 days. At expiration time a random parameter decides whether the customer renews its web hosting contract or not. The probability of renewal of the contract is set to 90% corresponding to the rate that is observed for the customers of our ISP. Server operation costs are withdrawn from the agents' accounts at the end of a year, and the return received for the hosting of services is promptly credited to the agents' accounts. The Allocator checks every 100 days whether it has made a loss of more than €10,000 during the past 200 days. If this is the case, the Allocator invests in a new data center in order to reduce the market prices of the resources and to return to a profitable operation of the business. The parametrization of $max_{DB}, max_{MAIL}, max_{WEB}$ relies on empirical values from the system's operation. The average consumption of bandwidth is set to 0.0005 MBit/sec per MByte web hosting capacity and to 0.001 MBit/sec per email account leased to the ISP's customers. The bandwidth capacity available to the system is acquired from an external service provider. The contingent of bandwidth available to the entire allocation system is increased stepwise if the load goes beyond 80% and is reduced if the load falls below 30%. The discount rate for the combinatorial package of web and mail resources is 5%. The profit margins of the agents $(g_{DC}, g_{FS}, g_{WEB}, g_{MAIL}, g_{DB})$ are dynamically

adapted: if an agent wins four consecutive bids, it increases the bid price by 5% and if, on contrary, an agent's bid is not successful the profit margin is reduced by 5%.

Fig. 7 shows the server load of the DCAgent, the profit of the FSAgent, and the price of a medium size web hosting package for a competitive system in the course of time of a simulation run (x-axis shows the day during a simulation run). The load curve of the DCAgent shows a strong decline at simulation day 200 and 400 (1). This results from investments into new server infrastructure that reduce the resource prices in the data center, which consequently leads to significantly reduced profits on the part of the service agents in the data center (2). In the long run the resource prices stabilize and the service agents return to profitable business (Fig. 7 lower part).

Fig. 8 depicts the results of the first simulation experiment. The experiment is designed to test the system's behavior in the monopoly situation depicted in Fig. 6. In this situation it is interesting to observe the profit distribution between the DCAgent and the Allocator. Because the Allocator has no alternative providers to choose from in the resource procurement process, it is obvious that the DCAgent can exploit the lack of competition while increasing its profit margin to the maximum level that is allowed in the allocation system. As a result of this exploitation the allocator permanently loses money whereas the DCAgents gains the total system profit. As a reaction to this disequilibrium, the production costs for the ISP's services remain at a level that is higher than the retail price for an average web hosting package.

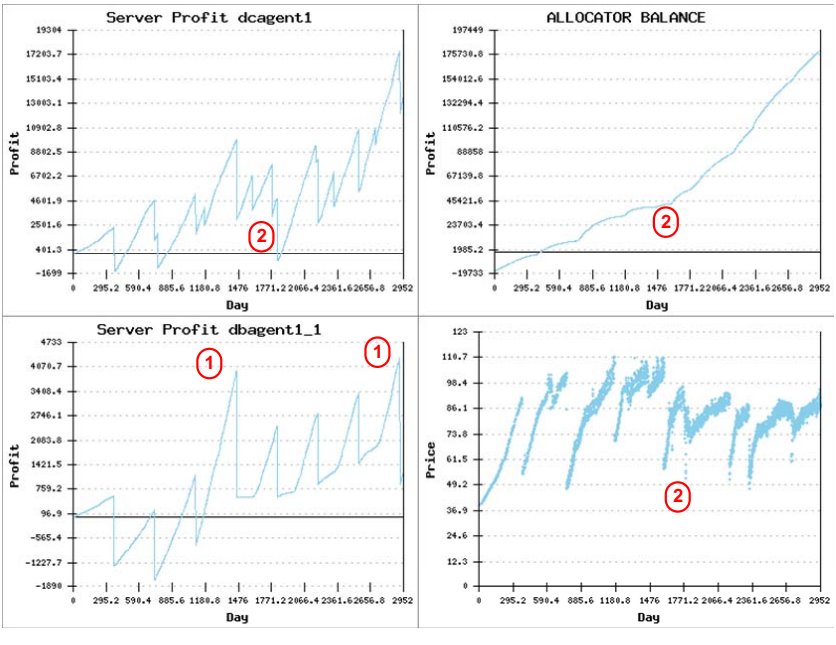


Fig. 9. Diagrams showing the profit of the DBAgent, the DCAgent, and the Allocator in the competitive situation and the resulting price for a medium size web hosting package in the course of time

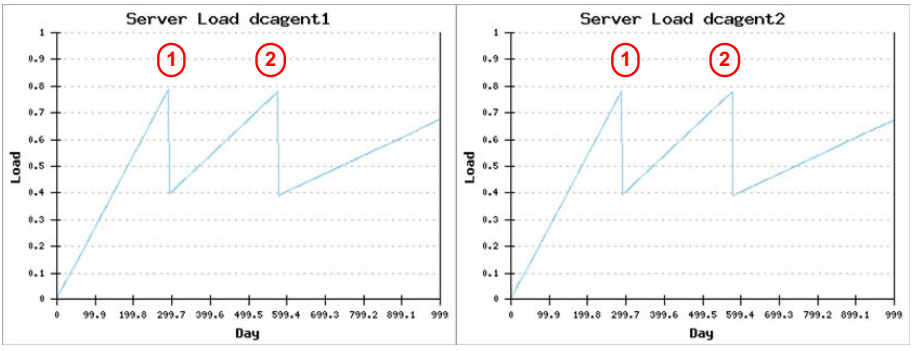


Fig. 10. Diagrams showing the server load reaction for DCAgents 1 and 2 in a situation that increases the bandwidth in the competitive system configuration

The second experiment depicted in Fig. 9 investigates the profit distribution between the DCAgent, the DBAgent, and the Allocator for the competitive situation. All agents are able to operate cost-covering in this simulation. The Allocator is able to accumulate a considerable profit over time. The profit of the Allocator represents the yield of the ISP company in our example. The DCAgent's profit is grows continuously while guaranteeing a budget for replacement costs at the end of its lifetime. Even the DBAgent, which usually has an unfavorable parametrization of its cost function in our example, is able to amortize its own investment at the end of its life cycle (1) after a short drop in (2). The production costs for the web hosting service are lower than the retail prices charged for an average web hosting package.

The third experiment is designed to demonstrate the system's load balancing capability. It is conducted in the competitive situation with two data centers involved. Fig. 10 shows the server load of DCAgent_1 and DCAgent_2. On simulation day 300 both servers reach the threshold load of 80%. This causes the DCAgents to double the available bandwidth capacity from 100 to 200 MBit/sec and reduces the server load to 40% (1). The process repeats at simulation day 576 (DCAgent_1) and 578 (DCAgent_2) (2). The system's reaction shows the functional capability of the load allocation system in a competitive situation.

6 Conclusions

We have presented an agent-based system that is designed for the automated allocation of web hosting services to the IT resources of a medium-sized ISP. The system should find a cost minimizing allocation of resources for web hosting services on the ISP's infrastructure. Agents assigned to the system's servers can independently determine the bid price for the resources they are responsible for. Together with a central allocator and a system of cost and price functions, we formulate an economic model that provides a load balancing function for the ISP's resource capacities. By performing experiments based on real world data we are able to demonstrate that the allocation system can guarantee a stable resource load and a 'fair' profit distribution if there is competition between the resource agents.

References

1. Altman, E., Boulogne, T., El-Azouzi, R., Jimenez, T., Wynter, L.: A survey on networking games in telecommunications. Comp. and OR 33(2), 286–311 (2004)
2. Cheng, J., Wellman, M.: The WALRAS Algorithm: A Convergent Distributed Implementation of General-Equilibrium Outcomes, Comp. Economics 12 (1996)
3. Davis, R., Smith, R.G.: Negotiation as a Metaphor for Distributed Problem Solving. Artificial Intelligence 20, 63–109 (1983)
4. Drexler, K.E., Miller, M.S.: Incentive Engineering: for Computational Resource Management. In: Huberman, B.A. (ed.) The Ecology of Computation, pp. 231–266. Elsevier, North-Holland (1988)
5. Eymann, T., Reinicke, M., Ardaiz, O., Artigas, P., Freitag, F., Navarro, L.: Decentralized Resource Allocation in Application Layer Networks. In: 3rd IEEE Int. Symposium on Cluster Computing and the Grid (CCGrid 2003), Tokyo, Japan, pp. 645–650 (2003)
6. Feldman, M., Lai, K., Stoica, I., Chung-I Chuang, J.: Robust Incentive Techniques for Peer-to-Peer Networks. In: Proc. 5th ACM Conf. on Electronic Commerce (EC 2004), pp. 102–111. ACM Press, New York (2004)
7. Ferguson D.: The Application of Microeconomics to the Design of Resource Allocation and Control Algorithms, Columbia University (1989)
8. Hansen, L., Hussla, I.: WEBHOSTS: A European Survey of Supply and Demand of Webhosting for SMEs. In: Proc. of the 9th Int. Conf. on Concurrent Enterprising (ICE), Espoo, Finland (2003)
9. Knaak, N., Kruse, S., Page, B.: An Agent-Based Simulation Tool for Modelling Sustainable Logistics Systems. In: Proceedings of the iEMSs Third Biennial Meeting: Summit on Environmental Modelling and Software. International Environmental Modelling and Software Society (2006)
10. Lai, K., Huberman, B.A., Fine, L.: Tycoon: A Distributed Market-based Resource Allocation System. Technical report, HP Labs, Palo Alto, CA, arXiv:cs.DC/0404013 downloaded 12/7/08 (2004)
11. Miller, M.S., Krieger, D., Hardy, N., Hibbert, C., Tribble, E.D.: An Automated Auction in ATM Network Bandwidth. In: Clearwater, S.H. (ed.) Market-based control: A Paradigm for Distributed Resource Allocation, pp. 96–125. World Scientific, Singapore (1996)
12. Schwind, M.: Dynamic Pricing and Automated Resource Allocation for Complex Information Services - Reinforcement Learning and Combinatorial Auctions. LNEMS, vol. 589. Springer, Berlin (2007)
13. Schwind, M., Stockheim, T., Rothlauf, F.: Optimization Heuristics for the Combinatorial Auction Problem. In: Proceedings of the Congress on Evolutionary Computation CEC 2003, Canberra, Australia, pp. 1588–1595 (2003)
14. Waldspurger, C.A., Hogg, T., Huberman, B.A., Kephart, J., Stornetta, S.: Spawn: A Distributed Computational Economy. Software Engineering 18(2), 103–117 (1991)

AgEx: A Financial Market Simulation Tool for Software Agents

Paulo André L. De Castro[1,2] and Jaime S. Sichman[2]

[1] Technological Institute of Aeronautics1, São José dos Campos, São Paulo, Brazil
pauloac@ita.br
[2] Intelligent Techniques Laboratory, University of São Paulo2, São Paulo, Brazil
jaime.sichman@poli.usp.br

Abstract. Many researchers in the software agent field use the financial domain as a test bed to develop adaptation, cooperation and learning skills of software agents. However, there are no open source financial market simulation tools available, that are able to provide a suitable environment for agents with real information about assets and order execution service. In order to address such demand, this paper proposes an open source financial market simulation tool, called AgEx. This tool allows traders launched from distinct computers to act in the same market. The communication among agents is performed through FIPA ACL and uses a market ontology created specifically to be used for trader agents. We implemented several traders using AgEx and performed many simulations using data from real markets. The achieved results allowed to test and assess comparatively trader's performance against each other in terms of risk and return. We verified that the effort to implement and test trader agents was significantly diminished by the use of AgEx. Furthermore, such results indicated new directions in trader strategy design.

Keywords: Autonomous agents, Software agents, Autonomous asset management.

1 Introduction

Many researchers have addressed the problem of creating mechanisms to automate the administration of assets. It is possible to observe the use of the most varied reasoning techniques, for instance: neural networks [1], reinforcement learning [2], [3], multiagent systems [4], [5], [6], BDI architectures [7], Case-based reasoning [8], SWARM approaches [9] and others [10]. These initiatives could be classified by their capability of handling several assets simultaneously (multi-asset) or with just a single asset (mono-asset). It should be pointed out that the administration of several assets is more complex than just to answer the administration of an asset. It is necessary to explore the complementarities among the group of assets, especially, to minimize the portfolio risk. Therefore, it is possible to classify the cited papers in two big groups: multi-asset [4], [5], [6], [9] and mono-asset [1], [2], [3], [7], [8], [10]. We verified that there are more papers in the second group (mono-asset) than in the first group. The reason for that becomes clear when we classify the works according with its agent

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 704–715, 2009.
© Springer-Verlag Berlin Heidelberg 2009

goal: profit maximization [1], [2], [3], [4], [5], [7], [8], [10] or some tradeoff solution between profit maximization and risk mitigation [6], [9]. It is clear that there is a much stronger concern in maximizing profit than in mitigating risks. This balance has also happened in the beginning of the study of portfolio administration and selection. One classic paper in Finance Theory that presented a significant contribution to risk measurement and control was presented by Markowitz [11] more than fifty years ago and it helped to change the exaggerated concern with profit against risk control. This concern shows that there is a lot to advance in order to achieve software agents that may be as efficient as human beings in portfolio management. One big obstacle to research in automated portfolio management is the need for a test bed for the designed agents and systems. This test environment should be able to simulate financial markets as close to reality as possible.

This kind of tool is fundamental to research in automated portfolio management, but it is not really part of it. It is an infrastructure that could be reused by a lot of researchers. This paper presents an open source financial market simulation tool with special features that makes it different from others tools currently available. This system is called AgEx (*Agent Exchange*) and it is available under LGPL license. Such license allows its free use by researchers, even in proprietary projects.

2 AgEx Architecture

Figure 1 presents the AgEx architecture with its main components and communications links among trader agents and their human investors. The gray rectangles represent software agents (traders, manager and broker), while the circles represent the owners of the agents and a human administrator of AgEx. The entity represented by a white rectangle is a software module that is too simple to be classified as agent, and performs the actions determined by agents, such as buy and sell order executions. The component AgEx Data is just a database of real operations that took place in some real exchange and it may be used in simulations as described later.

2.1 Main Components

The AgEx system is composed by three kinds of components:

- Trader Agent: It is responsible to decide and to submit buy or sell orders to some predefined assets. In fact, these agents use the AgEx as a simulation platform framework for communication and life cycle management. Therefore, they are represented over AgEx border in figure 1. The AgEx system may provide services for many traders simultaneously, as shown in figure 1.
- AgEx Manager: This agent is responsible to validate and to process messages addressed to AgEx system. It sends the valid messages to execution that are performed by a software module, called AgEx Broker. The execution results are received by the manager and sent to the traders that submitted the order.
- AgEx Broker: It receives and executes buy or sell orders and informs the AgEx Manager about the result of execution.

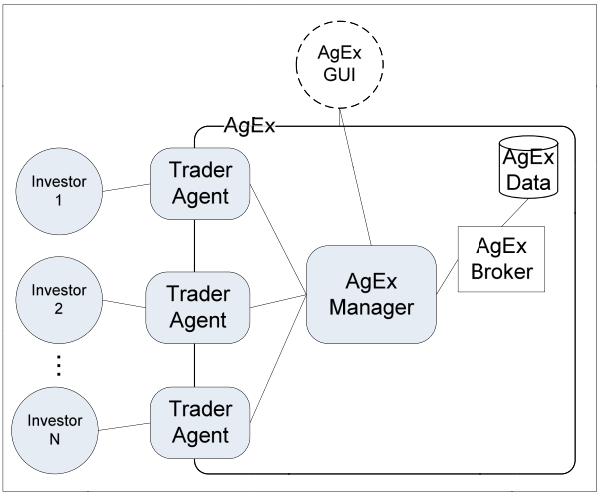


Fig. 1. AgEx Architecture and its main components and interface with external users and administrator

2.2 Communication and AgEx Ontology

The communication within the AgEx society (manager and traders) demands a way to interchange concepts and agent actions through messages. In order to address such demand, we developed a specialized ontology to AgEx based on a content reference model created by FIPA [12]. This ontology includes the main concepts, predicates and possible actions needed by trader agents, such as: asset, quote request, quote result, order submission, order result, etc. These concepts, predicates and actions are used to create content for any message exchanged within an AgEx society. The possible concepts in AgEx are *Order*, *MarketOrder*, *LimitedOrder*, *OrderResult*, *Query*, *QueryResult*, *AssetConcept*, *Error* and *Terminate*.

Each message in AgEx has a content entity extended from *ContentElement* class. This entity may be an action that can be performed by agent (derived from *AgentAction*). In this case, the message content is a request to the agent to perform one specific action. It is interesting to notice that an agent action is also a concept. The possible agent requests in AgEx are: *Order*, *MarketOrder*, *LimitedOrder* (traders submit orders for execution), *Query* (trader request information about assets) and Terminate (the manager tells the trader that the simulation is over). The AgEx Manager uses the derived classes of Result to inform the trader about the request results. When one trader submits an order execution request (*Order*, *MarketOrder* or *LimitedOrder*), it is responded with an *OrderResult* concept. Whether one trader submits a *Query* request, it is responded with a *QueryResult* concept. Whenever an invalid or unknown message is received by AgEx manager, it sends a message which content is an Error concept.

The communication among agents in AgEx is performed using the addressing and delivering services provided by the Java Agent Development Environment, JADE [13]. Such message service is compatible with the standards defined by FIPA [12].

The possible dialogs between the *AgExManager* and any trader agent are related to three specific situations, which are described next:

- Order submission: A trader agent decides to submit an order (buy or sell, limited or market). It creates a message with content equals to such order and sends it to AgExManager. This one returns a message with result of order execution.
- Query about some asset: The trader asks quote information about a specific asset. The AgExManager responds the message with the required information or an error message if the information is not available.
- Unknown or Invalid Messages: In case of the sent by the trader, the AgExManager responds with a message, which content is an Error concept.

2.3 Simulation Mechanism

In its default mode (historical or real price mode), AgEx allows simulation using asset information from real stock markets. This information is composed by assets prices (open, high, down and close prices) and volume (shares traded by assets). Therefore, the asset prices do not change according to trader orders: prices are defined by AgEx Data. This kind of simulation is particularly useful when the research is focused on the development of trading algorithms that do not account the effect caused by the own algorithm. In fact, this effect may be despised since the amount of assets traded by the agent is much smaller than the market volume.

However, researchers interested in analyzing the effect of some trader strategy in the market may use the second simulation mode in AgEx. It is called live price mode (or price formation mode). In live mode, the prices and volume are defined exclusively by the orders submitted from the trader agents.

The trader agents and the AgEx manager agent are synchronized by message exchanges. The manager defines the duration of each cycle (time step) and their transitions. All traders must be able to get the needed information, deliberate and submit orders within the interval of one time step. Whenever a trader doesn't complete these jobs within one time step, the system raises an overrun exception. One trader agent doesn't known in advance at which price one market order will be executed, just like it happens in real markets. Furthermore, agents are not allowed to access price information beyond the current cycle. These features provide more realism to the simulation and avoid that one trader gets privileged information.

2.4 Real Operation Mechanism

Despite the fact the AgEx is mainly concerned with simulation of financial markets, we designed it to be able to be used as a platform for software traders operation in real exchanges. This will be possible through the replacement of the AgEx Broker by another component that adapts the interface expected by the AgExManager to the interface provided by the target market exchange. This kind of component is called AgEx-Exchange Adapter. The development of an Adapter is not complex; however it requires an express permission from a real exchange or broker company in order to access the system. We intend to implement an example of an AgEx Adapter in the future.

3 AgEx Implementation

AgEx was implemented in Java using JADE platform and it is composed by more than 10.000 lines of code shared among 101 classes and interfaces. However, the development of a new trader agent requires the creation of only one class which must extend *AgExTraderAgent* class and implement one method responsible for order definition at each cycle. The other tasks demanded as get quote information, send orders to the market, compute results are performed by AgEx itself and may be configured through a specific configuration file.

In order to make easier the launcher of an agent society, AgEx provides a launcher that allows the definition of a society through XML files including parameters for traders (initial capital and traded assets, for instance). In figure 2, an example of such a society definition file is presented.

```
  <?xml version="1.0" encoding="UTF-8"?>
<society-agex>
  <manager  remote_manager="yes" hostname="127.0.0.1"
  port="1099"/>
 <trader name="RSI" initial_capital="1000000"
 path="agex.traders.RSI">
        <asset id="AAPL" initial_shares="0"/>
        <asset id="ADBE" initial_shares="0"/>
        <param  value="para1" />  </trader>
 <trader name="MA" initial_capital="1000000"
 path="agex.traders.MA">
  <asset id="AAPL" initial_shares="0"/> </trader>
</society-agex>
```

Fig. 2. An Example of society definition files with two trader agents. The manager tag defines that this society will be connected to an AgEx server running in the host indicated by its IP number and TCP port.

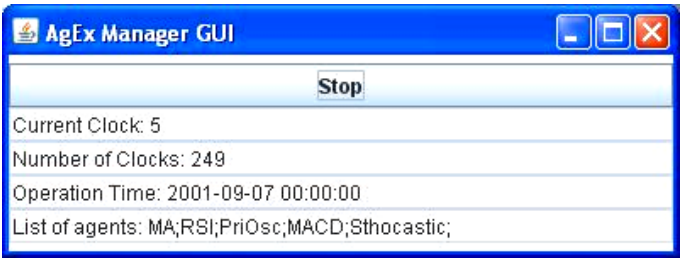


Fig. 3. AgEx Manager GUI

After launching a society, it is possible to follow its simulation progress or to pause it using a simple graphical user interface (figure 3). Furthermore, AgEx allows launching several societies from different computers at different times to the same market simulation (see figure 4). These societies are synchronized by the manager in order to observe the same simulation time.

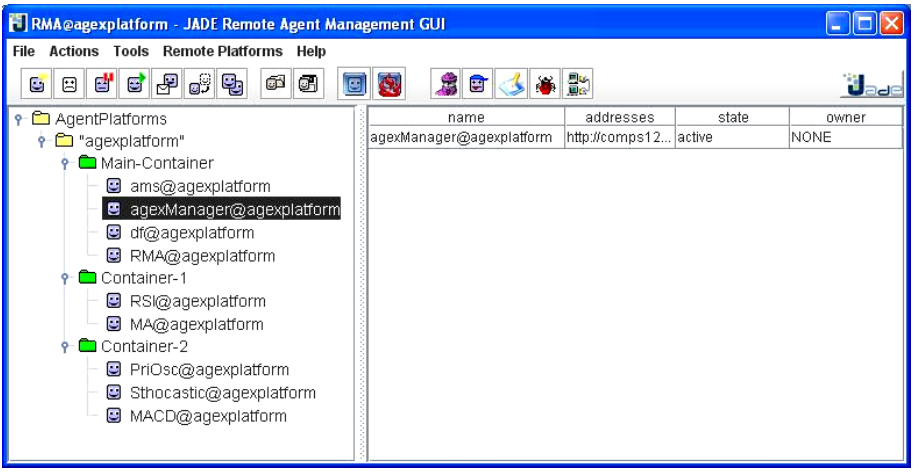


Fig. 4. AgEx trader agents distributed in three computers

In figure 4, we present an example of two societies launched from different computer in JADE Management GUI (in JADE, each container is associated to one computer), that are associated to AgEx manager in a third computer (Main-container). In the first container (Container-1), there are two traders agents (RSI and MA), while the second container has three agents (PriOsc, Sthocastic and MACD). All five trader agents deal with the agexManager agent located at the Main-container. The strategies used to build such trader agents are described later.

3.1 Simulation Generated Data

AgEx registers in the end of each cycle the position of each trader (money, shares, stock prices and orders). Furthermore, it creates a summarized file with the results of all traders in the computer that runs the Manager. These files are created in csv format that facilitates their analysis with spreadsheet programs (like Excel or Open Office).

3.2 Importing Data

Real quote information is essential to perform simulation of markets and also to provide data to agents that trade in real markets. Fortunately, several web sites (like Yahoo Finance, for instance) provide this kind of information free of charge. This information must be inserted into AgEx Data to be used by the system. AgEx Data is implemented as a Firebird RDBMS. We created a GUI to import quote information files with Yahoo Finance format. It makes easier the capture and information update in AgEx tool.

4 Related Work

In this section, we present a comparative analysis of some selected systems with similar propose of AgEx. Such analysis is based on some features that allow or

facilitate the simulation of markets to test and assess trader agents. We do not intend to judge the overall quality of the cited systems, but just identify differences (positive and negative) with the system proposed here.

The selected systems for analysis are eAuctionHouse [14], eMediator [15], PXS [8], SFI [16] and JASA [17]. In table 1, we present a comparative analysis of these systems based on four features.

The two first features (real and live price modes) were already discussed in the Simulation Mechanism section. The third feature indicates if the system source code is available free of charge. The fourth feature tells whether the system defines or uses ontology to exchange information (concepts or requests). AgEx is the only one that fulfills all features. In fact, it is the only open source tool able to perform historical price simulations. Additionally, we may say that AgEx is the only system that is adherent to FIPA recommendation for communication among agents. Despite the fact some may say that this is not clearly an advantage, we may argue that the adherence to standard communication patterns makes easier its use by others researchers.

Table 1. Comparison among Selected Systems

System	eAuctionHouse	eMediator	PXS	SFI	JASA	AgEx
Real Price mode	No	No	Yes	No	No	Yes
Live Price Mode	Yes	Yes	Yes	Yes	Yes	Yes
Open Source	No	No	No	Yes	Yes	Yes
Use ontology	No	No	No	No	No	Yes

5 AgEx in Action

We used AgEx platform in many simulations (sections 5.2, 5.3 and 5.4) in order to analyze five trader strategies based on technical indicators: *RSI*, *Price Oscillator (PriOsc)*, *Moving Average (MA), Moving Average Convergence-Divergence (MACD)* and *Stochastic*. Such indicators are widely used by financial analysts as part of their decision process. These strategies were implemented as single trader agents and each one took less than 150 lines of Java code: this indicates that AgEx really reduces the implementation effort of trader agents. We are not going to detail such strategies in this paper because they are explained in detail in the indicated references [6], [18].

In order to assess the trader's performance against the performance of their assets, we developed another trader agent that simply buy and hold one unit of each asset that it manages. This agent (*BuyAndHold*) is useful to give information about the evolution of asset prices. It's expected that a good trading strategy overcome the buy and hold strategy.

5.1 Experimental Setup

Frequently, studies on automated asset management present very limited experimental evaluation, for instance using one or very few assets and/or short evaluation periods. Another concern should be to avoid unclear selection criteria of assets and periods to avoid bias in such selection. These problems may cause wrong conclusions and

dangerous generalizations about the agent's performance in periods and assets that were not taken into account.

We tried to avoid this problem by selecting a long period (19 years) and several assets: stocks of 14 companies from 5 different economic sectors (technology, healthcare, services, consumer goods and apparel stores). We selected companies from Nasdaq 100 Index, which lists the 100 more relevant companies on Nasdaq Exchange.

Unfortunately, many companies listed are relatively new and therefore they don't present long temporal price series. In fact, there are only than 10 companies with at least 20 years of history. We preferred to reduce one year in the period, in order to get 14 assets with available price series of 19 years. These assets and companies are presented on table 2.

Table 2. Selected Stocks

ID	Name	ID	Name
AAPL	Apple Inc	DELL	Dell Inc
ADBE	Adobe Sys. Inc	INTC	Intel Corporation
ALTR	Altera Corp	JAVA	Sun Microsystems
AMAT	Applied Materials Inc.	MSFT	Microsoft Corp.
AMGN	Amgen Inc	ORCL	Oracle Corp.
CMCSA	Comcast Corp	PCAR	PACCAR Inc.
COST	Costco Wholesale Corp.	ROST	Ross Stores Inc.

5.2 Risk and Return Performance

We have performed simulations of six trader agents (*RSI, MACD, MA, PriOsc, Stochastic, BuyAndHold*) over the period of Jan 1, 1989 until Dec 31, 2007, where each agent were able to trade with 14 stocks (listed on table 2). The obtained results are presented in terms of annual return and risk (measured as standard deviation of agent's patrimony), in figure 5 and 6, respectively. These results show that there is no trader that overcomes the others in a consistent way over the whole period. In fact, several traders are replacing each other in the position of best performance as time evolves. This is true also when analyzing the traders under risk criteria. Tables 3 and 4 present these results clearer. In terms of final return, *RSI* obtained the best performance in six years and the second best in two years. However, we may see that *MACD* has very similar overall performance, because it achieved the best performance in others five years, the second best performance in two years and furthermore it got the third best in four years against only one of *RSI* trader. The others traders presented inferior results, but all got the best performance in some year except the *Buy and hold* trader. Therefore, we may conclude that there is no strong superiority of any analyzed traders regarding the final return. Probably, one trader strategy that uses a mix of the analyzed strategy could get better results. Furthermore, the poor performance of *Buy and Hold* trader makes it clear that is possible to achieve good results using active strategies.

Table 3. Final return results achieved by traders. Traders are sorted in alphabetical order. The ranking is defined by the number of times the trader achieved first, second or third places.

Trader	*Ranking*	*1o.*	*2o.*	*3o.*
Buy And Hold	4	3	6	6
MA	2	5	2	1
MACD	5	1	3	4
PriOsc	1	6	1	1
RSI	6	0	5	5
Sthocastic	3	4	2	2
Total	**-**	**19**	**19**	**19**

Table 4. Final risk results achieved by traders

Trader	**Ranking**	**1o.**	**2o.**	**3o.**
Buy And Hold	2	4	6	5
MA	5	2	7	2
MACD	3	4	0	4
PriOsc	6	0	3	2
RSI	1	6	2	1
Sthocastic	4	3	1	5
Total	**-**	**19**	**19**	**19**

In table 4, we may realize that traders with good performance according return criteria achieved this result at cost of higher risk. One may observe that RSI (first in return) has become the last in risk evaluation and MACD (the second in return) was just third in risk evaluation. Moreover, Buy and Hold trader (the sixth in return) is the second in risk evaluation. This performance inversion is not a surprise. In fact, it is compatible with the common notion that in order to achieve higher returns, it is necessary to accept higher risks.

5.3 Broker's Fee Influence

One common assumption in autonomous traders design is that as fees will be charged from all traders no matter its strategy, then strategies could be designed and compared among themselves without concern about fees, because they would reduce profitability of all traders in an almost equal way. The AgEx supports fees collection (a fixed amount by operation and/or a percentage of transaction volume). Therefore, we used this feature to verify this common assumption. We repeated the scenario described in section 5.2, but charging 10$ dollars and 0.5% of order volume (shares times price) per each order. Despite the fact, fees change very much among different brokers, these values may be considered typical. The summarized results obtained by trader agents for the same period and asset set used in section 5.2 is presented in table 5. In fact, achieved results showed that there is profitability reduction, but some agents were more affected than others. Observing table 5, one may realize that *Buy And Hold* agent is in better position than in table 3. This happened because this agent

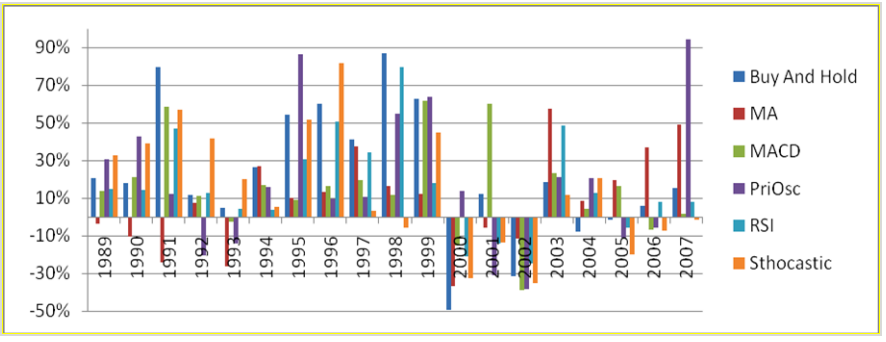


Fig. 5. Trader Agent's Return by year

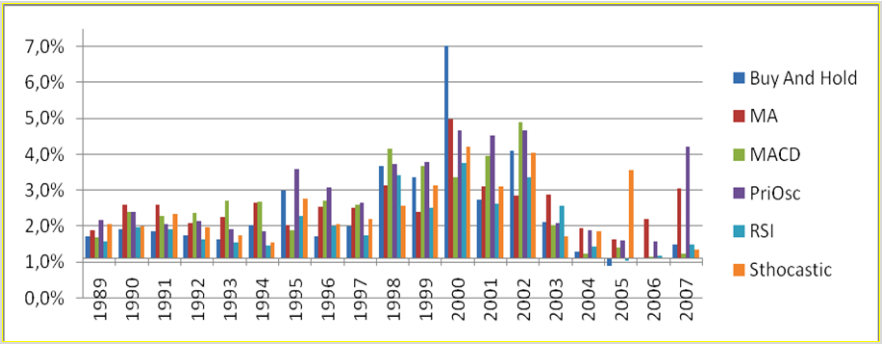


Fig. 6. Trader Agent's Risk. The risk is assessed as the standard deviation of agent's returns.

submits fewer orders than others so it paid fewer fees. One trader may benefit from submitting fewer orders, but each order with higher volume.

5.4 Trader Performance by Paper

Figure 7 presents daily average performance achieved by the trader agents for each paper. These results are obtained through simulation over the period from 2003 to 2007; each trader was allowed to deal with one asset. We realized that one agent with very good performance for an asset may get very poor results in another. For instance, the *RSI* agent was the first for AAPL and in the same period the fourth for AMGN.

Table 5. Final return results achieved by traders in simulation with fees

Trader	*Ranking*	*1o.*	*2o.*	*3o.*
Buy And Hold	2	4	6	5
MA	5	2	7	2
MACD	3	4	0	4
PriOsc	6	0	3	2
RSI	1	6	2	1
Sthocastic	4	3	1	5
Total	**-**	**19**	**19**	**19**

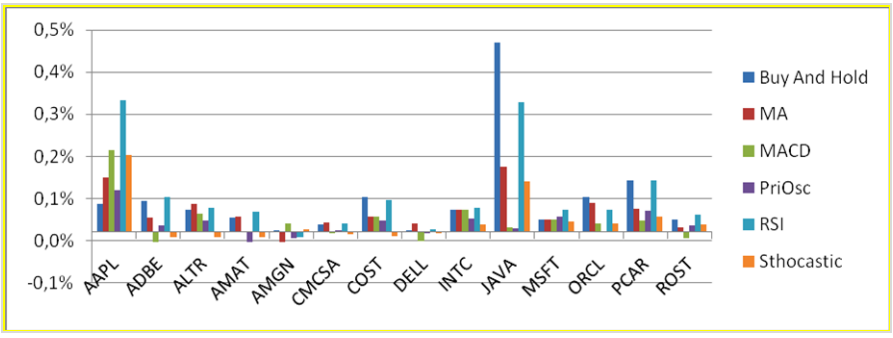


Fig. 7. Trader Agent's daily average return

6 Conclusions

The AgEx tool presented in this paper is a special-purpose software agent platform for simulation of financial markets. It is open source and allows market simulation with prices from real markets. It makes available a market ontology that simplifies communication. AgEx provides facilities to launch traders from several computers over the net and to analyze their performances. We have presented six trader agents implemented using AgEx and the obtained results from their simulation in several scenarios. In these implementations, we could realize that the effort to implement trader agents was significantly reduced by AgEx use. Furthermore, AgEx is adherent to international standards of agent communication [12]. This feature may facilitate its use by others researchers.

We have performed a significant amount of simulated experiments (over a period of 19 years, using 14 different assets) and tested the influence of broker's fee in trader performance. The obtained results for trader's performance were analyzed in terms of risk and return. The comparison among traders dealing with and without fees showed that the presence of fee may harm less one agent than others (section 5.3). The results also showed that there is no dominant strategy along the time (section 5.2) and no agent presented best performance for all papers (section 5.4) among analyzed traders.

Moreover, these analyses make us believe that new strategies mixing information from existing traders may achieve good results. We intend to use AgEx in our future research to develop this kind of trading strategy. Finally, we believe that AgEx can be very useful for others researchers trying to develop new strategies for automated asset management.

Acknowledgements. Jaime Sichman is partially supported by CNPq/Brazil.

References

1. Kendall, G., Su, Y.: Co-evolution of successful trading strategies in a simulated stock market. In: Proceedings of ICMLA 2003, Los Angeles, pp. 200–206 (2003)
2. Sherstov, A., Stone, P.: Three automated stock-trading agents: A comparative study. In: Proceedings of AMEC Workshop - AAMAS 2004, New York (2004)

3. Nevmyvaka, Y.F.Y., Kearns, M.: Reinforcement learning for optimized trade execution. In: Proceedings of the 23rd International Conference on Machine Learning- ICML 2006, Pittsburgh, Pennsylvania (2006)
4. Decker, K., Pannu, A., Sycara, K., Williamson, M.: Designing behaviors for information agents. In: Johnson, W.L., Hayes-Roth, B. (eds.) Proceedings of Agents 1997, pp. 404–412. ACM Press, New York (1997)
5. Luo, K.L.Y., Davis, D.N.: A multi-agent decision support system for stock trading. IEEE Network 16(1), 20–27 (2002)
6. Castro, P.A., Sichman, J.S.: Towards cooperation among competitive trader agents. In: Proceedings of 9th ICEIS. Software Agents and Internet Computing track, Funchal, Madeira - Portugal, pp. 138–143 (2007)
7. Feng, X., Jo, C.H.: Agent-based stock trading. In: Proc. of the ISCA CATA 2003, Honolulu, Hawaii (2003)
8. Kearns, M., Ortiz, L.: The penn-lehman automated trading project. IEEE Intelligent System 18(6), 22–31, 11-12 (2003)
9. Kendall, G., Su, Y.: A particle swarm optimisation approach in the construction of optimal risky portfolios. In: Proc. of the 23rd IASTED. Innsbruck, Austria, pp. 140–145 (2005)
10. Feng, R.Y.Y., Stone, P.: Two stock-trading agents: Market making and technical analysis. In: Proceedings of the Agent Mediated Electronic Commerce (AMEC) Workshop - AAMAS 2003, Melbourne, Australia (2003)
11. Markowitz, H.M.: Portfolio selection. Journal of Finance 7(1), 77–91 (1952)
12. FIPA. The Foundation for Intelligent Physical Agents, `http://www.fipa.org`
13. Bellifemine, F.L., Caire, G., Greendwood, D.: Developing Multi-Agent Systems with JADE. In: Wiley (ed.). Wiley Series in Agent Technology (April 2007)
14. Wurman, P.R., Wellman, M.P., e Walsh, W.E.: The Michigan Internet AuctionBot: A configurable auction server for human and software agent. In: AGENTS, Minneapolis/St. Paul, MN, pp. 301–308 (1998)
15. Sandholm, T.: eMediator: A Next Generation Electronic Commerce Server. In: International Conference on Autonomous Agents (AGENTS), Barcelona (June 2000)
16. LeBaron, B.: Building the Santa Fe Artificial Stock Market. Working Paper. Brandeis Univ. (2002)
17. Phelps, S.: Evolutionary Mechanism Design. Phd Thesis. Univ. of Liverpool (2007)
18. Market Screen Investment Tools website, `http://www.marketscreen.com`

A Domain Analysis Approach for Multi-agent Systems Product Lines

Ingrid Nunes[1], Uirá Kulesza[2], Camila Nunes[1],
Carlos J. P. de Lucena[1], and Elder Cirilo[1]

[1] Pontifical Catholic University of Rio de Janeiro (PUC-Rio)
Rio de Janeiro, Brazil
{inunes,cnunes,lucena,ecirilo}@inf.puc-rio.br
[2] Federal University of Rio Grande do Norte (UFRN)
Natal, Brazil
uira@dimap.ufrn.br

Abstract. In this paper, we propose an approach for documenting and modeling Multi-agent System Product Lines (MAS-PLs) in the domain analysis stage. MAS-PLs are the integration between two promising techniques, software product lines and agent-oriented software engineering, aiming at incorporating their respective benefits and helping the industrial exploitation of agent technology. Our approach explores the scenario of including agency features to existing web applications and is based on PASSI, an agent-oriented methodology, to which we added some extensions to address agency variability. A case study, OLIS (OnLine Intelligent Services), illustrates our approach.

Keywords: Multi-agent systems, Software product lines, Methodology, Web applications.

1 Introduction

Over the last years, agents have become a powerful technology to support the development of distributed complex applications. Software agents are a natural high-level abstraction that helps understanding and modeling this kind of systems. Agents usually present some particular properties [1], such as autonomy, reactivity, pro-activeness and social ability; therefore they facilitate developing systems that present autonomous behavior. This autonomy property refers to agents able to act without the intervention of humans or other systems: they have control both over their own internal state, and over their behavior [2]. Several methodologies [3,4,5] have been proposed in order to allow the development of Multi-agent Systems (MASs). However, most of them do not take into account the adoption of extensive reuse practices that can bring an increased productivity and quality to the software development.

Software product lines (SPLs) [6,7] have emerged as a new trend of software reuse investigating methods and techniques in order to build and customize families of applications through a systematic method. A SPL is defined as [7] "a set of software intensive systems that share a common, managed set of features satisfying the specific needs of a particular market segment or mission and that are developed from a common set of

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 716–727, 2009.
© Springer-Verlag Berlin Heidelberg 2009

core assets in a prescribed way". A feature is a system property that is relevant to some stakeholder and is used to capture commonalities or discriminate among products in a SPL. The main aim of SPL engineering is to analyze the common and variable features of applications from a specific domain, and to develop a reusable infrastructure that supports the software development. This set of applications is called a family of products.

The purpose of Multi-agent System Product Lines (MAS-PLs) is to integrate SPLs and MASs, incorporating their respective benefits and helping on the industrial exploitation of agent technology. In particular, the scenario we are currently exploring is the incorporation of autonomous or proactive behavior to existing web systems. The main idea is to introduce software agents into existing web applications in order to allow (semi)automation of tasks, such as autonomous recommendation of products and information to users. Due to the existence of many web applications already developed and deployed on application servers, our MAS-PL approach aims at extending these web applications with the aim to bring a minimum impact to their provided features and services that are adequately structured according to classical architectural patterns, such as Layer and MVC [8].

In this context, this work presents an approach for modeling MAS-PLs. We aim at proposing a methodology that covers the full domain engineering development process, which encompasses domain analysis, domain design and domain implementation. However, in this paper we focus at the first stage – domain analysis. In [9], we have identified particular kinds of variability of MAS-PLs and how effective SPL methodologies are for documenting them. We now propose an approach that extends PASSI [4], an agent-oriented methodology, to support managing SPL variabilities. PASSI provides a useful way for specifying a MAS, although it considers the development of single systems. We motivate and illustrate this work with the OLIS case study, a SPL that provides different personal services to users, such as calendar and events announcement.

The remainder of this paper is organized as follows. Some works related to MASs and SPLs are described in Section 2. In Section 3, an overview of the OLIS case study is presented, giving some details about its development. In Section 4, we show how we have modeled our SPL in the domain analysis stage, based on a PASSI extension. We present some discussions in Section 5. Finally, conclusion and directions for future work are discussed in Section 6.

2 An Overview of Existing SPL and MAS Approaches

Over the past few years, several methods have been published to address problems and challenges of SPL engineering. FORM [10] extended FODA [11] to cover the entire spectrum of domain and application engineering, including the development of reusable architectures and code components. Pohl et al. [6] propose a framework for SPL engineering that defines the key sub-processes of the domain engineering and application engineering process as well as the artifacts produced and used in these processes. PLUS [12] provides a set of concepts and techniques to extend UML-based design methods and processes for single systems in a new dimension to address SPLs. SPL methodologies provide useful notations to model the agency features; however, none of them

completely covers their specification [9]. Agent technology provides particular characteristics that need to be considered in order to take advantage of this paradigm.

On the other hand, many MAS methodologies have been proposed. Tropos [3] provides guidance for the four major development phases of application development (Early requirements, Late requirements, Architectural design and Detailed design). PASSI [4] brings a particularly rich development lifecycle that spans initial requirements though deployment and, in addition, emphasizes the social model of agent-based systems. Using the analogy of human-based organizations, Gaia [5] provides an approach that both a developer and a non-technical domain expert can understand. A particular objective of our study was to find out how these methodologies can be used to help on the MAS-PL development.

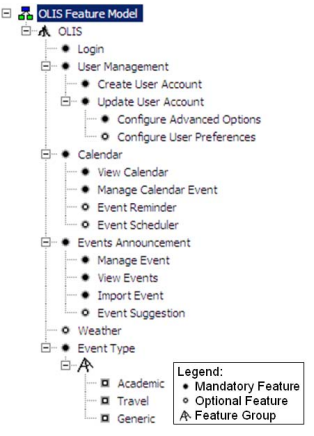
Recent research work has investigated the integration synergy of MASs and SPLs technologies. Dehlinger & Lutz [13] have proposed an extensible agent-oriented requirements specification template for distributed systems that supports safe reuse. Although it proposes useful templates, it addresses only one specific kind of agency variability: the intelligence level. Pena et al. [14] propose an approach that consists of using goal-oriented requirement documents, role models, and traceability diagrams in order to build a first model of the system, and later use information on variability and commonalities throughout the products to propose a transformation of the former models that represent the core architecture of the family. According to the approach, the variabilities are analyzed after modeling the MAS, and this can lead to undesired situations, such as, the high coupling between mandatory and optional features and inadequate modularization of agency features.

3 OLIS Case Study

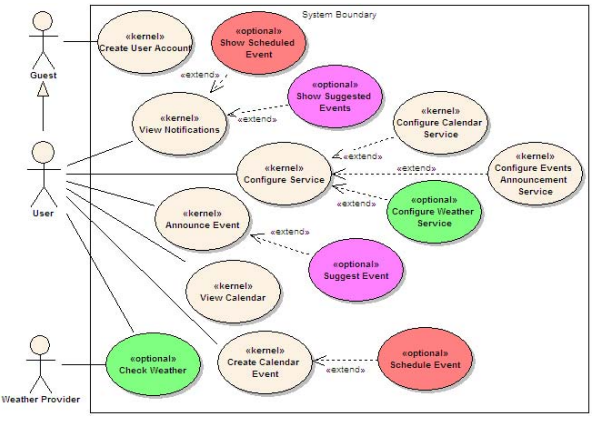
In our work, we have developed two MAS-PLs to drive our study and both followed the reactive SPL adoption strategy, which advocates the incremental development of SPLs. Both MAS-PLs represent initially web systems that are extended to incorporate new autonomous optional features. The first one is a SPL of conference management web systems, the ExpertCommittee [15], on which we added autonomous behavior. The second one, the OLIS [16], is detailed in this section. OLIS case study exploits the BDI (belief-desire-intention) model, which is supported by several agent platforms, such as Jadex. It is used to illustrate our approach in next section.

The OLIS (**On**-**L**ine **I**ntelligent **S**ervices) case study is a SPL of web applications that provide several personal services to users. The first version of the SPL is composed mainly of two services: (i) Events Announcement and (ii) Calendar services. The Events Announcement service allows users to announce events to other system users through an events board. The Calendar service lets users to schedule events in their calendar. Announced events can be imported to the users' calendar. OLIS was designed in such a way that the system can be evolved to incorporate new services without interfere in the existing ones. Additionally, the product line has an alternative feature: the type of event that it manages, which can be: generic events, academic events and travel events.

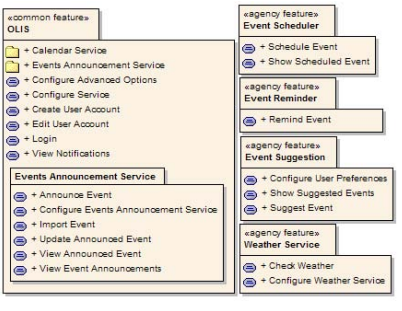
After developing the first version of OLIS product line, new autonomous behavior features were introduced to automate some tasks in the system. We evolved OLIS by



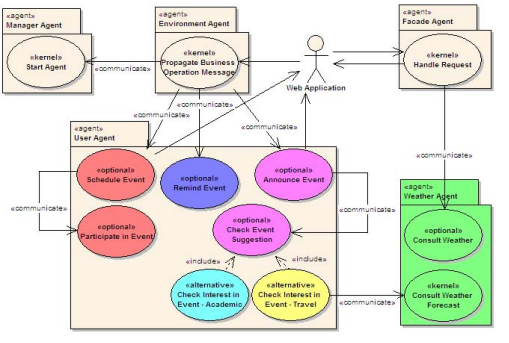
(a) Feature Model



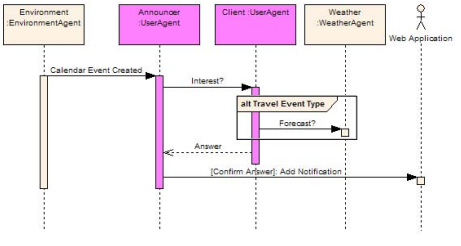
(b) Use Case Diagram



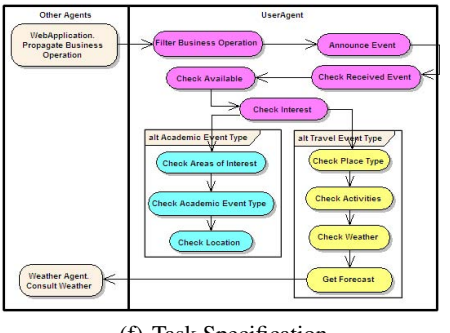
(c) Feature/Use Case Dependency model



(d) Agent Identification



(e) Role Identification



(f) Task Specification

Fig. 1. OLIS Diagrams

adding new features to it, which take advantage of the agents technology. The services become intelligent services. Figure 1(a) shows the feature model of the OLIS MAS-PL with its new optional features.

The new features incorporated to the OLIS product line were: (i) *Events reminder* – users configure how many minutes they want to be reminded before events, and the system sends messages to notify users about events about to begin; (ii) *Events scheduler* – when an user adds a new calendar event that involves more participants, the system checks the schedule of the other participants to verify if this event conflicts with other existing ones. If so, the system suggests a new date for the calendar event that is appropriate according to all schedules from participants; (iii) *Events Suggester* – when a new event is announced, the system automatically recommends the event after checking if it is interesting to users based on their preferences. This feature is also responsible for checking if the weather is going to be appropriate according to the place type where the event is going to take place; (iv) *Weather* – this is a new user service. It provides information about current weather conditions and forecast of a location. This service is also used by the system to recommend announced travel events.

The evolution of the OLIS MAS-PL was accomplished by the introduction of software agents and their respective roles on the architecture. Agents were implemented using JADE and Jadex platforms. There are five different agents types: (i) `Environment Agent` – perceives changes in the data model and propagates them to other agents; (ii) `FacadeAgent` – retrieves information from agents to business services, it is a facade between the web application and the agents; (iii) `WeatherAgent` – provides information about the weather and weather forecast; (iv) `ManagerAgent` – starts `UserAgents` when the system starts up or when a new user is inserted in the system; (v) `UserAgent` – each user has an agent that represents him/her in the system. It acts on behalf of the user. Each `UserAgent` is composed of roles, which implement agency features. For example, the `EventScheduler` and `EventParticipant` roles implement the *Events scheduler* feature. Due to space restrictions in this paper, we only briefly described OLIS architecture. For additional details of it, please refer to [17].

4 Modeling OLIS in Domain Analysis with a PASSI Extension

In this section, we present our approach for modeling MAS-PLs, which is based on the PASSI methodology. Due to the deficiencies and lack of expressivity of PASSI for documenting variability, we propose extensions to document agency features in MAS-PLs. We focus in this paper specifically on the domain analysis stage, yet the main aim of our work is to define a set of guidelines to model and document agency features along all SPL development stages. The case study previously presented (Section 3) is used to illustrate our extensions. It is important to notice that, although the need to clearly modeling agency features came from the incremental and reactive development of OLIS and ExpertCommittee, the extension proposed here can also be useful when adopting proactive and extractive development strategies.

PASSI (Process for Agent Societies Specification and Implementation) [4] is an agent-oriented methodology that specifies five models with their respective phases for developing MASs. The methodology covers all the development process, from requirements to code. The five different models from PASSI are System Requirements Model,

Agency Society Model, Agent Implementation Model, Code Model and Deployment Model. The domain analysis stage corresponds to the System Requirements Model, which generates a model of the system requirements in terms of agency and purpose. PASSI follows one specific guideline: the use of standards whenever possible; and this justifies the use of UML as modeling language. However, the UML semantics and notation is extended to design specific needs of agents. PASSI methodology is designed for developing single systems, therefore we had to adapt it to express variability.

4.1 Feature Modeling

Feature modeling is an important activity in SPLs, which is not covered by PASSI. It is the activity of modeling the common and variable properties of concepts and their interdependencies. The features are organized into a coherent model referred to as a feature model, which specifies the features of a SPL as a tree, indicating mandatory, optional and alternative features. Features are essential abstractions that both customers and developers understand. Originally, the feature model was proposed in [11].

Figure 1(a) illustrates the OLIS feature model. It shows its different kinds of features: (i) mandatory – features that are in all the versions of the system and are part of its core. Examples are the *Calendar* and *Events Announcement* features; (ii) optional – features that can be introduced when customizing specific versions of OLIS MAS-PL, such as the *Event Scheduler* and *Event Suggester* features; and (iii) alternative – features that varied from one version to another one. There are different types of events and one of them must be chosen in the product derivation process [18]. Besides the feature model, constraints express feature interdependencies. Some features can depend on another, e.g. the *Weather* feature must be present if the Event Suggester and Travel Event Type are selected, and some features are mutually exclusive, the event type illustrates this constraint.

4.2 Domain Requirements Description

According PASSI, in the Domain Requirements Description phase, we make a functional description of the system composed of a hierarchical series of use case diagrams. An important difference when modeling use cases for SPLs is that use cases must be decomposed into two or more use cases connected by relationships such as extend and include, if they have an optional or alternative part (see *Suggest Event* use case in Figure 1(b)). This is essential for providing feature modularization. Moreover, in order to enable variabilities modeling, we have adapted these PASSI diagrams using the PLUS method notation. In the PLUS approach, stereotypes are used to indicate if a use case is mandatory (kernel), alternative or optional. Figure 1(b) shows a partial view of the OLIS MAS-PL use case model. Besides stereotypes, we also colored use cases to indicate which feature they are related to. This indication is used in all artifacts to provide a better understanding of features traceability.

PASSI suggests describing use cases using sequence diagrams. However, we preferred to use the common use case descriptions to explain use cases. Use case descriptions are widely used in the literature, and are also adopted by PLUS.

In SPLs, it is very important to keep a traceability of the features along all artifacts generated while they are being modeled. So, it is necessary to model the dependency between features and use cases. Therefore, a diagram, which is inspired by PLUS, showing another view of use cases to express this dependency relation must be generated: use cases are grouped into UML packages that represent features. These packages are stereotyped with: (i) <<*common feature*>> - represents all mandatory features and groups all kernel use cases; (ii) <<*optional feature*>> - represents optional features and groups use cases related to a specific optional feature; (iii) <<*alternative feature*>> - represents alternative features and groups use cases related to a specific alternative feature; and (iv)<<*agency feature*>> - indicates that use cases of a specific package are related to an agency feature. The first three stereotypes are used combined with the last one, meaning that an agency feature can be part of the SPL kernel, or an optional or alternative feature.

Figure 1(c) illustrates a partial view of the OLIS Feature/Use Case Dependency model. It shows a package stereotyped with <<*common feature*>>, which contains the kernel use cases, and packages stereotyped with <<*agency feature*>> and <<*optional feature*>> because these use cases are related to optional agency features.

4.3 Agent Identification

The input of this phase are the use case diagrams generated in the Domain Requirements Description phase. Responsibilities are attributed to agents, which are represented as stereotyped UML packages. The OLIS Agent Identification Diagram is depicted in Figure 1(d).

PASSI methodology considers that all functionalities of the system are performed by agents. Agents are entities that usually presents autonomy and pro-activeness, so only functionalities that have these characteristics need to be performed by agents. Thus, only use cases that are into an UML package stereotyped with <<*agency feature*>> will be considered to be performed by an agent. These use cases will be grouped into <<*agent*>> stereotyped packages so as to form a new diagram. Each one of these packages defines the functionalities of a specific agent.

In OLIS case study, each user of the system has an agent that represents it. This agent is an instance of the `UserAgent`. We did not find how to express the communication between different instances of the same agent in PASSI, so we drew an arrow between two use cases that goes outside the UML package and then comes back (see Figure 1(d)).

4.4 Role Identification

In the Role Identification phase of PASSI, all the possible paths (a "communicate" relationship between two agents) of the Agents Identification diagram are explored. A path describes a scenario of interacting agents working to achieve a required behavior of the system. In different scenarios, agents can play different roles. Agent interactions are expressed through sequence diagrams.

Usually, each scenario corresponds to only one feature of the system. For these cases, a feature dependency table to map sequence diagrams to each feature is enough.

However, there are some features that can impact other ones. In these cases, we say that the feature crosscuts the other. An example is the *Event Suggestion* feature of OLIS, which is an optional agency feature. Its behavior varies according to which event type is chosen. When an event is inserted on the system, the `UserAgent` that represents the user who inserted it asks for other user agents if they have interest on that event. So, the agent checks the user availability on the event date. Besides, if the event type is academic, the agent checks the areas of interest and the location of the event according to the user `AcademicPreferences`. And if the event type is travel, the agent checks the place type where the event is going to happen and the activities that can be done according to the user `TravelPreferences`; and the agent also consults the `WeatherAgent` to get the weather forecast and checks if it will be good in the event date. Thus, according to the *Event Type* feature, the agents behavior changes. The solution that we found for this problem is the use of UML 2.0 frames to express optional and alternative paths. Figure 1(e) illustrates this scenario. In this diagram, only the interaction among agents/roles are reported, the internal behavior of agents are specified in the next phase.

4.5 Task Specification

In the Task Specification phase, activity diagrams are used to specify capabilities of each agent. According to PASSI, for every agent in the model, we draw an activity diagram that is made up of two swimlanes. The one from the right-hand side contains a collection of activities symbolizing agent's tasks, whereas the one from the left-hand side contains some activities representing other interacting agents.

In these diagrams, we have made three adaptations, some of them were already adopted in other diagrams: (i) instead of drawing only one diagram per agent, we split the diagram according to features; (ii) use of UML 2.0 frames to show different paths when there is a crosscutting feature; (iii) a colored indication showing with which feature the task is related to.

These adaptations can be seen in Figure 1(f). According to PASSI, the `UserAgent` should have only one task specification diagram, but in the presented diagram there are only activities related to the *Event Suggestion* agency feature. In addition, the *Event Type* feature crosscuts this feature, so we have an UML 2.0 frame indicating alternative activities according to the event type (academic or travel). Finally, activities are colored indicating that they are related to *Event Suggestion*, *Academic* and *Travel* features. Note that these are the same colors used in previous diagrams.

4.6 PASSI Adaptations - Overview

The main objective of PASSI adaptations proposed in our work is to provide a better feature modularization and traceability, including new agency features that need to be addressed in MAS-PLs. Splitting diagrams in the way we proposed allow the selection of the necessary diagrams during the application engineering according to selected features. Nevertheless, crosscutting features could not be isolated from the others, and this is a challenge that we are still facing while developing MAS-PLs (see Section 5).

Table 1. PASSI extensions

Phase	Extensions
New phase: Feature Modeling	
Domain Requirements Description	Use of Stereotypes (kernel, alternative or optional) Refactoring use cases to modularize features Feature/Use Case Dependency Model Use of colors to trace features
Agent Identification	Only use cases in <<agency feature>> stereotyped packages are distributed among agents Use of Stereotypes (kernel, alternative or optional) Use of colors to trace features
Role Identification	Use of UML 2.0 frames (crosscutting features) Use of colors to trace features
Task Specification	One diagram per agent and feature Use of UML 2.0 frames (crosscutting features) Use of colors to trace features

Several PASSI extensions were proposed to model agency variabilities. Most of them came from PLUS [12] approach, which provides useful notations to model SPLs [9]. Table 1 summarizes the adaptations we proposed.

The proposed adaptations provide important advantages for modeling MAS-PLs. Feature modeling is a fundamental phase on which the commonalities and variabilities of the SPL are analyzed. The main purpose of other adaptations (e.g. use of UML 2.0 frames and use of colors) is to allow the features modularization and traceability along all the artifacts of the domain analysis stage. This is essential in SPLs because their variable parts are changed or removed in an easier way. Besides, the approach proposed here of using the agent abstraction to model only the agency features allows taking advantage of other technologies. This is better discussed in next section.

5 Discussions

In this section, we present and discuss some lessons learned while modeling agency features in MAS-PLs and challenges that we still have to face. These lessons learned offer directions for a methodology for developing MAS-PLs that we are currently defining.

Integration of SPL techniques with existing Multi-agent Methodologies. Several MAS methodologies have been proposed [3,4,5]. These methodologies have the purpose of only supporting the development of agent-based systems; however each has its own unique perspective and approach to developing MASs. Though, no one methodology is useful in every system–development situation. One question that we had to deal while modeling MAS-PL is what methodology should be our start point. PASSI was chosen because it integrates concepts from object-oriented software engineering and artificial intelligence approaches. It uses an UML-based notation, and it brings facilities to the incorporation of notations already proposed for SPLs [12]. In our approach, we explore the synergy of integration of these both approaches.

Explicit Separation of Modeling and Implementation of MAS Features from other Technologies. MAS methodologies usually propose to distribute all system functionalities/responsibilities among agents. Agents are an abstraction that provides some particular characteristics, such as autonomy and pro-activeness. In our approach, we adopted the <<*agency feature*>> stereotype to indicate features that present autonomous behavior and should be modeled using agent abstraction. Besides, we claim that features

of the SPL that do not take advantage from agent technology can be modeled and implemented using existing SPL approaches. In our work, this explicit separation contributes to facilitate the mapping of the MAS-PL domain analysis models to domain design and implementation artifacts, in the following way: (i) agency features can be separately modeled and subsequently mapped to MAS design notations and agent implementation frameworks (e.g. Jade and Jadex); (ii) non-agency features can take benefit from other existing SPL methodologies and technologies adopted to implement the variabilities.

Granularity in MAS-PLs. In the literature, there are many examples of SPLs with coarse-grained features. This means that these features can be implemented wrapped in a specific unit, such as a class, a method or an agent. Although, fine-grained extensions, e.g. adding a statement in the middle of a method, usually require the use of techniques, like conditional compilation, which obfuscates the base code with annotations. Though many SPLs can and have been implemented with the coarse granularity of existing approaches, fine-grained extensions are essential when extracting features from legacy applications [19]. Our scope in this study was dealing with coarse-grained features; however we are currently extending OLIS MAS-PL by adding new fine-grained features to explore this scenario. Our preliminary results show that the use of aspect-oriented techniques is fundamental to completely modularize the fine-grained agency features at both modeling and implementation levels.

Crosscutting agency features. Many of the agency features are implemented by a set of different system components, agents and classes. They are characterized as crosscutting features, because their design and implementation are typically spread and tangled along different system modules. Our approach does not provide clear support to deal with the documentation of these crosscutting features, however we are currently investigating how existing aspect-oriented modeling approaches [20,21] can help the visual documentation of the agency features. Studies about agency features modularity [22] using aspect-oriented programming have already reported some results in this direction.

6 Conclusions and Future Work

In this paper, we presented an approach for modeling MAS-PLs at the domain analysis stage. Our approach is based on PASSI methodology, which supports the specification of software agents. We have extended this methodology to address agency variabilities in product lines. An important phase that needs to be added in the methodology is the feature modeling, which is the activity that identifies SPL common and variable features. In addition, we have extended the PASSI notation, using stereotypes to indicate the variable abstractions and components of the systems. Since PASSI is based on the UML notation, it allowed us to adopt notations from PLUS, an existing SPL approach. We also discussed some important topics that arose from our study, such as the need of a clear support for crosscutting features and the use of object-oriented techniques in agent-based applications in order to allow the extension of existing web applications to incorporate agency features.

Our focus in this paper was in the domain analysis stage, but we are currently working on the development of a methodology that allows an explicit documentation and tracing of agency features throughout the SPL development process. Some SPL

methodologies are not used in practice due to their high complexity. Thus, we aim at developing an agile and adaptable methodology. Our methodology is being organized as a process framework composed of: (i) a core – that defines a set of mandatory activities and artifacts; and (ii) specific customizations – that specify additional activities and artifacts to the core according to specific scenarios that need to be addressed. Tool support for the methodology based on model-driven engineering techniques is also under development. Finally, we are also exploring in our methodology the definition of explicit mapping rules between analysis and design models and implementation artifacts that facilitates the code generation of MAS-PL architectures in existing implementation frameworks. Research work [23] that promotes pattern reuse in PASSI methodology is also being investigated in this sense.

References

1. Wooldridge, M., Ciancarini, P.: Agent-Oriented Software Engineering: The State of the Art. In: Ciancarini, P., Wooldridge, M.J. (eds.) AOSE 2000. LNCS, vol. 1957, pp. 1–28. Springer, Heidelberg (2001)
2. Wooldridge, M.: Intelligent Agents. The MIT Press, London (1999)
3. Bresciani, P., et al.: Tropos: An agent-oriented software development methodology. In: AAMAS 2004, vol. 8(3) (2004)
4. Cossentino, M.: IV. In: From Requirements to Code with the PASSI Methodology. Idea Group Inc. (2005)
5. Wooldridge, M., et al.: The gaia methodology for agent-oriented analysis and design. In: AAMAS 2000, vol. 3(3) (2000)
6. Pohl, K., et al.: Software Product Line Engineering: Foundations, Principles and Techniques. Springer, Heidelberg (2005)
7. Clements, P., Northrop, L.: Software Product Lines: Practices and Patterns. Addison-Wesley, USA (2002)
8. Buschmann, F., et al.: Pattern-Oriented Software Architecture: A System of Patterns. John Wiley & Sons, Chichester (1996)
9. Nunes, I., et al.: Documenting and modeling multi-agent systems product lines. In: SEKE 2008, pp. 745–751 (2008)
10. Kang, K., et al.: FORM: A feature-oriented reuse method with domain-specific reference architectures. Ann. Softw. Eng. 5 (1998)
11. Kang, K., et al.: Feature-oriented domain analysis (FODA) feasibility study. Technical Report CMU/SEI-90-TR-021, SEI, Carnegie-Mellon University (1990)
12. Gomaa, H.: Designing Software Product Lines with UML: From Use Cases to Pattern-Based Software Architectures. Addison Wesley, USA (2004)
13. Dehlinger, J., Lutz, R.R.: A Product-Line Requirements Approach to Safe Reuse in Multi-Agent Systems. In: SELMAS 2005. ACM Press, USA (2005)
14. Peña, J., Hinchey, M.G., Ruiz-Cortés, A., Trinidad, P.: Building the core architecture of a multiagent system product line: with an example from a future nasa mission. In: Padgham, L., Zambonelli, F. (eds.) AOSE VII / AOSE 2006. LNCS, vol. 4405, pp. 208–224. Springer, Heidelberg (2007)
15. Nunes, I., et al.: Developing and Evolving a Multi-Agent System Product Line: An Exploratory Study. LNCS. Springer, Heidelberg (2009) (to appear)
16. Nunes, I., et al.: Extending web-based applications to incorporate autonomous behavior. In: WebMedia 2008, pp. 115–122 (2008)

17. Nunes, I.: Towards a multi-agent product line development methodology (2008), http://www.inf.puc-rio.br/~ioliveira/maspl/
18. Cirilo, E., et al.: A Product Derivation Tool Base on Model-Driven Techniques and Annotations. JUCS 14, 1344–1367 (2008)
19. Kästner, C., et al.: Granularity in software product lines. In: ICSE 2008 (2008)
20. Jacobson, I., Ng, P-W.: Aspect-Oriented Software Development with Use Cases (Addison-Wesley Object Technology Series). Addison-Wesley, Reading (2004)
21. Clarke, S., Baniassad, E.: Aspect-Oriented Analysis and Design: The Theme Approach (The Addison-Wesley Object Technology Series). Addison-Wesley, Reading (2005)
22. Nunes, C., et al.: On the modularity assessment of aspect-oriented multi-agent systems product lines: a quantitative study. In: SBCARS 2008 (2008)
23. Cossentino, M., et al.: Patterns reuse in the PASSI methodology. In: Omicini, A., Petta, P., Pitt, J. (eds.) ESAW 2003. LNCS, vol. 3071. Springer, Heidelberg (2004)

A Reputation-Based Game for Tasks Allocation

Hamdi Yahyaoui

Information and Computer Science Department
King Fahd University of Petroleum and Minerals, Dhahran 31261, Saudi Arabia
hamdi@kfupm.edu.sa

Abstract. We present in this paper a distributed game theoretical model for tasks allocation. During the game, each agent submits a cost for achieving a specific task. Each agent, that is offering a specific task, computes the so-called reputation-based cost, which is the product between the submitted cost and the inverse of the reputation value of the bidding agent. The game winner is the agent which has the minimal reputation-based cost. We show how the use of reputation allows a better allocation of tasks with respect to a conventional allocation where there is no consideration of the reputation as a criteria for allocating tasks.

Keywords: Game, Reputation, VCG, Tasks Allocation.

1 Introduction

With the advent of Internet-based technologies, there is a growing interest in the design and implementation of distributed solutions. The emergence of distributed cooperation in networks is one of the current IT trends. This is exemplified by the tremendous success of several tools like Kaza, Gnutella, etc. The management of these systems is generally done in an ad-hoc way and does not rely on solid foundations that allow an optimal use of resources. Tasks allocation is one of the challenging research issues. This issue is seen as an optimization problem that can be solved using several mathematical modelling techniques.

Mechanism design is one of these techniques. It is a promising research field that aims at studying the aggregation of private preferences of self-interested agents in order to fulfill a social function [14]. This field gathers knowledge from game theory and economics [12]. The marriage of these two disciplines is behind the emergence of several centralized [15] and distributed [8] incentive-based solutions for solving several optimization problems.

In this paper, we present a distributed reputation-based game theoretical model for tasks allocation. During the game, each agent submits a cost for achieving a specific task. Each agent, that is offering a specific task, computes the so-called reputation-based cost, which is the product between the submitted cost and the inverse of the reputation value of the bidding agent. The game winner is the agent which has the minimal reputation-based cost. The contributions of this paper are threefold:

– We provide a distributed reputation-based solution for the tasks allocation problem, which is based on mechanism design.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 728–736, 2009.
© Springer-Verlag Berlin Heidelberg 2009

- We prove that our reputation-based solution is better than a allocation where reputation is not included from a task completion likelihood point of view.
- We provide some experiments that show the impact of considering reputation as a parameter in performing tasks allocation.

The rest of the paper is organized as follows. Section 2 is dedicated to the presentation of some preliminaries on mechanism design. In Section 3, we discuss the related work. Section 4 is devoted to the presentation of our reputation-based game for task allocation. Section 5 outlines the proof of concept and the related results. Finally, we draw some conclusions in Section 6.

2 Preliminaries on Mechanism Design

The general setting of a mechanism design problem is as follows [18,8]: There are n agents, each of them has a kind of private preference t_i, which is called also type. An output function maps each type vector $t = (t_1, t_2, \ldots, t_n)$ to a set of outputs. Each agent i has a valuation function v_i, which assigns a real number $v_i(t_i, o)$ to the output o. A mechanism defines for each agent i a set of strategies A^i. For each input $(a_1, a_2, \ldots, a_n)$, the mechanism computes an output $o = o(a_1, a_2, \ldots, a_n)$ and a payment vector $(\pi_1, \pi_2, \ldots, \pi_n)$ with $\pi_i = \pi_i(a_1, a_2, \ldots, a_n)$. Each agent i tries to maximize his own selfish utility $u_i(t_i, o) = v_i(t_i, o) + \pi_i$. The payment can be seen as an incentive for agents to reveal their true preferences.

A strategyproof mechanism is one in which types are part of the strategy space A^i and where each agent maximizes his utility by revealing his type t_i. This means that $\forall i, t_i, a_{-i}, a_i$,

$$v_i(t_i, o(a_{-i}, t_i)) + \pi_i(a_{-i}, t_i) \geq v_i(t_i, o(a_{-i}, a_i)) + \pi_i(a_{-i}, a_i) \quad (1)$$

Where a_{-i} denotes the vector of strategies of all the agents except i.

A direct-revelation mechanism [18,16] is a mechanism in which the only actions available to agents are to make direct claims about their preferences to the mechanism.

A famous direct-revelation mechanism that is meant to solve maximization of objective functions is the Vickery-Clark-Groves (VCG) mechanism. A direct-revelation mechanism $m = (o, \pi)$ belongs to the VCG family if

1. $o(t) \in argmax_o(\sum_{i=1}^{n} v_i(t_i, o))$
2. $\pi_i(t) = \sum_{i \neq j} v_j(t_j, o(t)) + h_i(t_{-i})$

Where $h_i(t_{-i})$ is an arbitrary function of t_{-i}.

It was already proven that a VCG mechanism is truthful, i.e., direct-revelation and strategy-proof mechanism. The idea of VCG is to choose the offer, which has the minimal cost and pay the winner with the second least offer. By doing so, the agents are constrained to reveal their true types.

A weighted implementation of VCG is formulated as follows [15]: A direct-revelation mechanism $m = (o, \pi)$ belongs to the weighted VCG family if

1. $o(t) \in argmax_o(\sum_{i=1}^{n} v_i(t_i, o))$
2. $\pi_i(t) = \frac{1}{\beta_i} \sum_{i \neq j} v_j(t_j, o(t)) + h_i(t_{-i})$

Where β_1,β_2,. . .,β_n are strictly positive numbers and the objective function is $g(o,t) = \sum_i \beta_i \, . \, v_i(t_i, o(t))$. In this paper, we reuse the established results about the weighted VCG family.

3 Tasks Allocation Problems

Tasks Allocation is defined as an assignment of a set of machines to a set of tasks together with an objective function. The aim is to find a feasible schedule optimizing the objective function. This problem was studied from a game theory point of view. It is defined as a game between agents in which each agent follows some specific strategy to win the game. The aim is to have a situation where no agent has incentive to unilaterally change his strategy. This situation is characterized as a Nash Equilibrium.

There are several game theoretical models that were proposed in the literature and that can be applied to tasks allocation such as the KP model [11] where the agents are considered as tasks and the objective function of an agent i is to minimize the completion time of the machine on which task i is executed.

In the CKN model [4], the objective function of an agent i is more the completion time of task i. In the AT model [1], the agents are considered as uniform machines and the objective function of each agent i is to maximize the profit defined as p_i-w_i/s_i, where p_i is the payment given to machine i, w_i is the load of machine i, and s_i is the speed of machine i. In [2], the authors consider an environment is which each user is selfish and has the goal to minimize the makespan of her own tasks. They model this problem as a non-cooperative, extensive-form game. They use the subgame perfect equilibrium solution concept to analyze the game which provides insight into the problem's properties.

In [7], the authors designed a new trust-based mechanism for tasks allocation. The work does not extend the VCG mechanism since the major work assumption is that agents reveal reputations of other agents and so in the paper problem setting the information about reputation is public rather than private.

We follow in this paper a different strategy where agents rely on their direct experiences rather than recommendations from other agents. These values are kept secret and can be updated by the agent that is offering a specific task upon the successful or unsuccessful execution of a task. We consider that agents are honest in such update. This led to the smooth extension of VCG and hence to the preservation of the theoretical results of that mechanism.

Our intent is to provide a reputation-based game in which the objective is to assign tasks to agents in a way that maximizes the likelihood of performing successfully these tasks. The reputation is a measure of what past collaborations with an agent tell about his performance. During the proposed game, each agent submits a cost for achieving a specific task. Each agent, that is offering a specific task, computes the so-called reputation-based cost, which is the product between the submitted cost and the inverse of the reputation value of the bidding agent. The game winner is the agent which has the minimal reputation-based cost and it is paid the second minimum like what is done in VCG auctions. We show how the use of reputation allows a better allocation of tasks with respect to a conventional allocation where there is no consideration of the reputation as a criteria for assigning tasks.

4 Reputation-Based Tasks Allocation Problem

We present in this section a formulation of our task allocation problem. We introduce a set of notations that will be followed in the rest of the paper. Let:

- $\mathcal{A} = \{A_1, A_2, \ldots, A_n\}$ be a finite set of agents and
- $\mathcal{T} = \{T_1, T_2, \ldots, T_m\}$ be a finite set of tasks.

We suppose that $\forall T_i \in \mathcal{T}$, T_i is indivisible and non-sharable. We assume that we have a finite sequence of tasks that should be done and we would like to assign an agent to each task of the sequence. Each assigned task will be tagged with a natural number that indicates its position in the sequence. We consider a projection p on a set of tuples, which returns the set composed by first elements of these tuples. The problem is to find a allocation $\mathcal{S} : \mathcal{A} \rightarrow \mathcal{P}(\mathcal{T} \times \mathcal{N})$, with $\mathcal{S}(A_i) \cap \mathcal{S}(A_j) = \emptyset, \bigcup_{A_i \in \mathcal{A}} p(\mathcal{S}(A_i)) = \mathcal{T}$ and $\mathcal{N}$ the set of natural numbers, which minimizes the total allocation cost, i.e., the following weighted utilitarian social welfare function:

$$f(rc, \mathcal{S}) = \sum_{k=1}^{n} \sum_{(i,j) \in \mathcal{S}(A_k)} rc_{k,i} \tag{2}$$

Where

$$rc_{k,i} = 1/r_{k,i} \,.\, v_{k,i} \tag{3}$$

In the above equation, $r_{k,i}$ denotes the reputation of the agent A_k in achieving the task T_i and $v_{k,i}$ denotes the cost of achieving the task T_i by the agent A_k. $rc_{k,i}$ is called the *reputation-based cost* for achieving the task T_i. The reputation is built through past experiences of assignment of tasks and will be discussed later. We are looking for minimizing the social choice function f, i.e., the objective for our mechanism to implement is:

$$\min \textstyle\sum_{k=1}^{n} \sum_{(i,j) \in \mathcal{S}(A_k)} rc_{k,i}$$

In order to make the agents reveal their true preferences, payments are introduced in a weighted VCG setting. The payment of an agent A_k for achieving a task T_i is defined as follows.

$$\pi_{k,i} = r_{k,i} \,.\, \sum_{(i,j) \in \mathcal{S}(A_k), k' \neq k} \frac{1}{r_{k',i}} v_{k',i} \tag{4}$$

Hence, in order to maximize his own utility, the agent should maximize his payment, which is equivalent to give the true cost of achieving a certain task. Concretely, a payment can be a disk storage, memory space, internet connection, etc.

The reputation of an agent is assessed by the agent offering a specific task. The latter keeps track of agent performance regarding achieving or not a specific task T. The reputation value is updated after each game round in which one specific agent is assigned a specific task of the tasks sequence. This value is set to a default value between 0 and 1 (different from 0 as stated in the weighted VCG problem formulation)

if the agent, that is offering a specific task, finds that it is the first time that an agent is asked to perform the task T. This is a simple bootstrapping strategy. Other strategies to solve such issue can be also adopted. The reputation value can be promoted if the agent successfully performs a task and it can be demoted otherwise. We consider the agents honest in such update. The reputation update strategy we adopt is as follows:

- Demotion:

$$R^{new} = \frac{R^{old}}{R^{old} + 1} \tag{5}$$

- Promotion:

$$R^{new} = 2 \times \frac{R^{old}}{R^{old} + 1} \tag{6}$$

In Equations 5 and 6, R^{new} and R^{old} denote respectively the new and old reputation values of an agent. It is clear that the reputation values in these equations are in the interval [0,1].

It is worth to mention that the more an agent has a good reputation the less he got a reputation-based cost. This means that it is in the interest of an agent to have a good reputation in achieving the tasks that are assigned to him.

To assess the added value of our reputation-based strategy, we define the overall reputation of the sequence of the tasks as the product of the reputation of all the agents, which have been assigned to specific tasks. More formally, the overall reputation is defined as follows:

$$\mathcal{R}_{\mathcal{A},\mathcal{S}} = \prod_{A_k \in \mathcal{A}} \prod_{(i,j) \in \mathcal{S}(A_k)} r_{k,i} \tag{7}$$

The use of products in the definition of the overall reputation is done in a way that a very small agent reputation value has a big impact on the result so that its effect will not be mitigated by a higher value if used in a summation.

It is expected that the overall reputation value in the designed game is better than the one resulted from a game that does not take the reputation as a criteria to assign tasks to agents. To verify that, it is sufficient to note that for any selected agent A_k and assigned task T_i that has the order j in the tasks sequence and such that $(i, j) \in \mathcal{S}(A_k)$, we can easily prove that $r_{k,i}$ is the highest reputation among all the reputations of the bidding agents.

Theorem 1. *The overall reputation value defined in equation 7 is better than any other reputation value deduced from a VCG-like game.*

Proof. Assume that the winner of the reputation-based game is the agent A_k. This means that $\forall k' \neq k.\ \frac{1}{r_{k,i}} v_{k,i} \leq \frac{1}{r_{k',i}} v_{k',i}$. Now assume that the winner of the game without reputation is the agent $A_{k'}$ with $k' \neq k$. This means that $v_{k',i} \leq v_{k,i}$. Hence, we get $\frac{1}{r_{k,i}} \leq \frac{1}{r_{k',i}}$ and finally we conclude that $r_{k,i} \geq r_{k',i}$.

5 Experiment

We consider a game including 6 agents that compete to win one of 4 tasks and a matrix 6×4, that tracks the reputation of each agent reputation in achieving each of the 4 tasks.

Table 1. Reputation matrix of agents

	T1	T2	T3	T4
A1	0.74	0.92	0.81	0.74
A2	0.83	0.99	0.56	0.78
A3	0.88	0.15	0.77	0.99
A4	0.98	0.58	0.15	0.53
A5	0.72	0.98	0.67	0.42
A6	0.77	0.76	0.28	0.35

Table 2. Tasks cost matrix

	T1	T2	T3	T4
A1	12	72	96	28
A2	93	60	54	54
A3	93	3	8	81
A4	81	12	2	46
A5	38	32	56	64
A6	70	25	45	67

Table 3. Sequence of tasks that should be performed

T2.T4.T2.T3.T4.T1

Table 4. Results of the reputation-based game without promotion and demotion

A3 is assigned T2 in round 1 with the bid: 3.0
A1 is assigned T4 in round 2 with the bid: 28.0
A3 is assigned T2 in round 3 with the bid: 3.0
A3 is assigned T3 in round 4 with the bid: 8.0
A1 is assigned T4 in round 5 with the bid: 28.0
A1 is assigned T1 in round 6 with the bid: 12.0

If the agent was never assigned a certain task T_k then its reputation value is default value (different from zero) for that task. This condition is meant to solve the bootstrapping issue that is faced in reputation systems [13]. We did an experiment with the reputation matrix specified in Table 1 (randomly generated reputation values).

The cost matrix is specified in Table 2. The sequence of tasks that should be achieved is specified in Table 3. The game is done several rounds and the result of each round is specified in Table 4. We don't consider for the moment promotion and demotion of reputation after each round. After the game is done, the resulted assignment matrix is specified in Table 5.

According to matrix 5, the overall reputation value is 0.007. If we assume that the game is done without considering reputation and with the same parameters, we get the assignment matrix that is shown in Table 7. The overall reputation is 0.001, which is clearly less than what we get using our reputation-based strategy.

Table 5. Allocation matrix that is result of the reputation-based game without promotion and demotion

	T1	T2	T3	T4
A1	1	0	0	1
A2	0	0	0	0
A3	0	1	1	0
A4	0	0	0	0
A5	0	0	0	0
A6	0	0	0	0

Table 6. Results of the game without reputation

A3 is assigned T2 in round 1 with a bid 3.0
A1 is assigned T4 in round 2 with a bid 28.0
A3 is assigned T2 in round 3 with a bid 3.0
A4 is assigned T3 in round 4 with a bid 2.0
A1 is assigned T4 in round 5 with a bid 28.0
A1 is assigned T1 in round 6 with a bid 12.0

Table 7. Allocation matrix for the game without reputation

	T1	T2	T3	T4
A1	1	0	0	1
A2	0	0	0	0
A3	0	1	0	0
A4	0	0	1	0
A5	0	0	0	0
A6	0	0	0	0

Table 8. Allocation matrix that is result of the reputation-based game with promotion and demotion

	T1	T2	T3	T4
A1	1	0	0	1
A2	0	0	0	0
A3	0	1	1	0
A4	0	1	0	0
A5	0	0	0	0
A6	0	0	0	0

Now, if we consider the promotion and demotion of reputation after each round of the game and we assume that agent A_3 failed in performing T_2 in round 1 and that agent A_1 succeeded in performing T_4 in round 2. The reputation of A_3 for T_2 is demoted as described in Section 4 and it becomes 0.13 instead of 0.15. While the reputation of A_1 for T_4 is promoted and it becomes 0.85 instead of 0.74. This makes T_2 assigned to A_4 instead of A_3 in round 3 with the bid 12.0 but A_1 still wins T_4 in round 5 since his reputation is higher than the one in the game without promotion and demotion. The new allocation matrix is the following:

6 Conclusions

We provided in this paper a distributed reputation-based game solution for tasks allocation. During the game, each agent submits a cost for achieving a specific task. The agent, that is offering a specific task, computes the so-called reputation-based cost. The game winner is the agent, which has the minimal reputation-based cost. We have shown how the use of reputation allows a better allocation of tasks with respect to a conventional allocation where there is no consideration of the reputation as a criteria for assigning tasks. Our future work will include the study of the impact of agent recommendations on the game. A challenge will be the design of a mechanism that allows to have a direct-truthful revelation or in the worse case a mechanism where liars, i.e., agents giving false reports have small impact on the equilibrium.

Acknowledgements. The author would like to express his thanks and appreciation for the support provided by KFUPM in the preparation of this paper.

References

1. Archer, A., Tardos, E.: Truthful Mechanisms for One-Parameter Agents. In: Proceedings of the 42nd IEEE Symposium on Foundations of Computer Science (FOCS), Washington, USA, pp. 482–491 (2001)
2. Carroll, T., Grosu, D.: Selfish Multi-User Task Scheduling. In: Proceedings of The Fifth International Symposium on Parallel and Distributed Computing, pp. 99–106 (2006)
3. Chevaleyre, Y., Dunne, P.E., Endriss, U., Lang, J., Lemaître, M., Maudet, N., Padget, J., Phelps, S., Rodríguez-Aguilar, J.A., Sousa, P.: Issues in Multiagent Resource Allocation. Informatica 30(1), 3–31 (2006)
4. Christodoulou, G., Koutsoupias, E., Nanavati, A.: Coordination Mechanisms. In: Automata, Languages and Programming: 31st International Colloquium, Turku, Finland, pp. 345–357 (July 2004)
5. Conitzer, V., Sandholm, T.: Complexity of Mechanism Design. In: Proceedings of the Uncertainty in Artificial Intelligence Conference (UAI), Edmonton, Canada, pp. 103–110 (2002)
6. Conitzer, V., Sandholm, T.: Automated Mechanism Design for a Self-interested Designer. In: Proceedings of the 5th ACM conference on Electronic Commerce, New York, USA, pp. 132–141 (2004)
7. Dash, R., Ramchurn, S., Jennings, N.: Trust-Based Mechanism Design. In: Third International Joint Conference on Autonomous Agents and Multiagent Systems, New York, USA, pp. 748–755 (2004)
8. Feigenbaum, J., Papadimitriou, C., Sami, R., Shenkar, S.: A BGP-based Mechanism for Lowest-Cost Routing. Distributed Computing 18(1), 61–72 (2005)
9. Feigenbaum, J., Papadimitriou, C., Shenker, S.: Sharing the Cost of Multicast Transmissions. JCSS: Journal of Computer and System Sciences 63(1), 21–41 (2001)
10. Feigenbaum, J., Shenker, S.: Distributed Algorithmic Mechanism Design: Recent Results and Future Directions. Bulletin of the European Association for Theoretical Computer Science 79, 101–121 (2003)
11. Koutsoupias, E., Papadimitriou, C.: Worst-case equilibria. In: Meinel, C., Tison, S. (eds.) STACS 1999. LNCS, vol. 1563, pp. 404–413. Springer, Heidelberg (1999)
12. Mas-Colell, A., Whinston, M., Green, J.R.: Microeconomic Theory. Oxford University Press, Oxford (1995)

13. Maximilien, E., Singh, M.: Reputation and Endorsement for Web Services. SIGecom Exchanges 3(1), 24–31 (2002)
14. Nisan, N.: Algorithms for Selfish Agents. In: Proceedings of the Annual Symposium on Theoretical Aspects of Computer Science, Trier, Germany, pp. 1–15 (1999)
15. Nisan, N., Ronen, A.: Algorithmic Mechanism Design. In: Proceedings of ACM Symposium on Theory of Computing, Atlanta, USA, pp. 129–140 (1999)
16. Nisan, N., Ronen, A.: Computationally Feasible VCG Mechanisms. Journal of Artificial Intelligence Research (JAIR) 29, 19–47 (2007)
17. Osborne, M.J., Rubenstein, A.: A Course in Game Theory. MIT Press, Cambridge (1994)
18. Parkes, D.C.: Iterative Combinatorial Auctions: Achieving Economic and Computational Efficiency. PhD thesis, Department of Computer and Information Science, University of Pennsylvania (2001)
19. Sandholm, T.: Automated Mechanism Design: A New Application Area for Search Algorithms. In: Rossi, F. (ed.) CP 2003. LNCS, vol. 2833, pp. 19–36. Springer, Heidelberg (2003)

Remote Controlling and Monitoring of Safety Devices Using Web-Interface Embedded Systems

A. Carrasco, M. D. Hernández, M. C. Romero, F. Sivianes, and J. I. Escudero

Dpto. Tecnología Electrónica, Universidad de Sevilla
Avda. Reina Mercedes S/N, 41012 Sevilla, Spain
{acarrasco,mdhv,mcromero,fsivianes,ignacio}@us.es

Abstract. To date, access control systems have been hardware-based platforms, where software and hardware parts were uncoupled into different systems. The Department of Electronic Technology in the University of Seville, together with ISIS Engineering, have developed an innovative embedded system that provides all needed functions for controlling and monitoring remote access control systems through a built-in web interface. The design provides a monolithic structure, independence from outer systems, easiness in management and maintenance, conformation to the highest standards in security, and straightforward adaptability to applications other than the original one. We have accomplished it by using an extremely reduced Linux kernel and developing web and purpose-specific logic under software technologies with an optimal resource use.

Keywords: Embedded systems, web applications, open source, safety devices, web interface, remote controlling.

1 Introduction

Nowadays, physical access control systems play a key role in corporation's security systems nowadays. They allow intrusion detection and grant access to authenticated users to facilities, devices and other sensible elements located in corporations. Their core is formed by hardware elements such as sensors, and other higher-level devices that allow their management.

The main drawback of this approach lies in the need of updating every single copy of the administration software each time a new update is released. Moreover, there is a strong dependence between the software and the hardware platform it runs on, entailing at worst the replacement of hardware parts to support new software features.

The main goal we proposed when designing the device described within this paper was designing from the scratch, and implementing, a new device that allows monitor and administer an access-control system remotely. This access-control [1] system consisted of multiple hardware parts monitoring unauthorized entrance to rooms within a corporation facility. The system had to surpass the previously commented drawbacks and finally, the device should be easily plugged into any Ethernet network offering its management features through a web-based interface.

Another objective, but not less important, was building a highly secure system.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 737–744, 2009.
© Springer-Verlag Berlin Heidelberg 2009

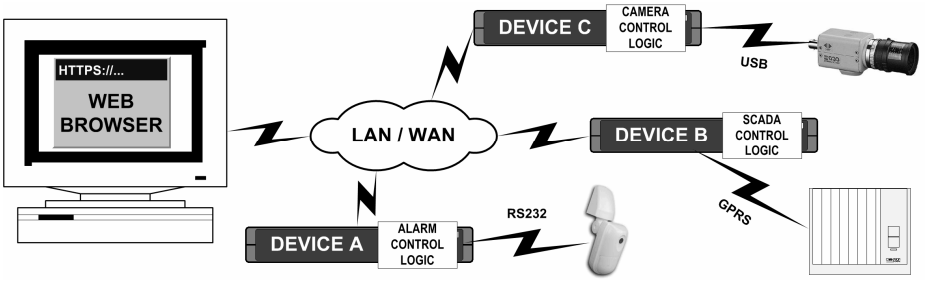


Fig. 1. Using the web-based embedded device for different applications

However, on a long term, the goal was to design a flexible system that could easily be adapted to other uses than monitoring access-control systems, with the least possible modifications to its hardware and software structures.

Apart from the obvious benefits enjoyed by final users, from the developer point of view, having a flexible platform implementing all the hardware parts and interfaces reduces development effort, thus reducing costs.

On the other side, convergence of software and hardware into a single device brings many advantages not only to access-control-orientated systems, but also to other specific-purpose systems such as Supervisory Control and Data Acquisition (SCADA) [3] systems, to give an example.

The solution to all these requirements comes by hands of an embedded platform [2] that implements software logic and hardware interfaces into a single device. The embedded system uses a hardware platform, which together with a modularized design of software components, allow adaptation to new applications.

2 Key Features of the System

Next we will discuss in detail the most noticeable features offered by the embedded device, both to final users and to developers themselves.

2.1 Compatibility

Using a web-based interface provides full software compatibility with any external operating system. Users can operate the device trough a web navigator application, available in all modern operating systems, (using PCs, cell phones…).

On the other hand, having a device based on Ethernet technology with a fully customizable IP makes integrating it into any corporative network a very straightforward process.

2.2 Monolithic Structure

One of the main goals achieved by this platform is to be designed in a way users perceive it like a *black box*. The system, once connected to the hardware it is designed to work with, allows interoperating with the subjacent hardware through the

web-based, built-in administration logic, so that users do not need to know the way devices lying below this platform work.

In a practical case, the box is connected to a Texecom's "Premier" alarm control panel through a serial port, having an Ethernet [4] port so that it is accessible from a Local Area Network and/or Internet. Users, using a web-browser, will connect to a given IP address and receive graphical information about the state of surveyed locations, performing any needed actions unaware of the panel and sensors connected to it.

2.3 Security

Security requirements for parts used in these applications (e.g. access control systems [5]) are very tight, because any security flaw in its software components might compromise the integrity global system.

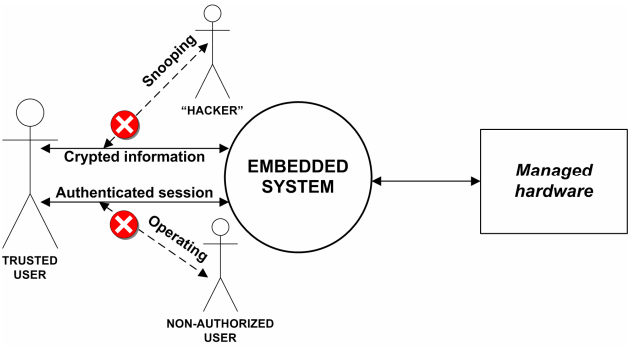


Fig. 2. Securing the embedded system

Therefore, some actions have been taken to ensure that only trusted users can make use of the embedded device's capabilities.

Firstly, information sent and received by our system is secured by means of Secure Socket Layer (SSL) encrypted HTTP connections [6] for all web communications. This means anybody is able to view any sensible information.

Furthermore, user authentication is implemented, which prevents non-authorized users from accessing to the embedded device. All users and groups management is administered from the web-interface; this way, users can be granted to view and/or modify different device's functions depending on the role they play within the corporative organization (e.g. administrators, maintainers, etc…).

Finally, additional securing steps have been taken, as the closure of all unessential and vulnerable network ports (e. g. Telnet [7])[8].

2.4 Ease in Management and Maintenance

Web logic provides a graphical interface that allows any user to use a web navigator to supervise the status of all the devices. It can also be used to deal with all other appropriate system operations such as bypassing of locks, deactivation of alarms, modification of parameters (e.g. the box's IP [9] address), or even updating the

system logic. Being able of updating system software logic from a remote computer is one of the strongest points of this architecture; this introduces many advantages:

- Final users do not need to update or change any software neither any hardware component in their systems, so they can use new features from a first moment.
- Technical assistance and maintenance can be carried out from remote computers, thus physical presence is avoided and maintenance expenses are severely reduced.
- In the case of a system hardware failure, it can be easily replaced by a new one, again without the need of any additional update at users´ environments.
- The option exists of making the system to automatically check for updates (by asking administrators for authorization or by using a fully automatic mode- update itself).

3 System Design

When designing software for embedded systems, we are subject to serious restrictions due to the limited system resources available, such as the processor clock frequency, RAM, and ROM. Other major parameters for the design of the overall system are the processor's power consumption and the cost of the processor. At the same time, there is an increased demand to improve the software-based functionality in the individual device.

Hardware used to build the embedded system consists of all-in-one board having:
- A RISC processor
- 2 MB of flash memory and 8 MB of RAM memory
- Two 100-base-t Ethernet ports, with unique MAC [10] addresses
- 3 serial ports: two RS-232 ports and one USB [11] port

This set-up provides a limited but powerful-enough platform to develop any networked device that can easily adapt to any purpose; its functionality can range from acting as a network firewall to -by simply updating its software logic- be used for communicating with any device via its serial or network ports.

The system can also easily be expanded to add different ports, as e.g. RS-485 or GPRS links [12].

3.1 Operating System

The system is built over a Linux [13] core, which due to the limited hardware resources available, needs to have a very small size. So, we have used some solutions to reduce the size of the embedded Linux operating system:

- All unneeded drivers and services have been removed.
- Only essential shell commands are kept, with reduced functionality. Furthermore,
- they have been put together into a common executable.
- O.S. is also compressed when stored into the flash memory, being decompressed to RAM memory to be effectively executed.

Linux daemons such as HTTP/FTP servers, security suites (firewall, SSL, etc.) or custom device controllers are also deployed as a part of Linux operating system, which are upon booting the system.

3.2 Software Components Design

Due to low spare flash space and RAM memory available, custom software components also need to be small both in size and memory use.

The best approach here is using software components developed in pure C language. Also, server-side scripting languages such as PHP [14] or ASP [15] cannot be used in such an environment, leading us to use client-side JavaScript language and Common Gateway Interface (CGI) [16] executable programs to build the web logic.

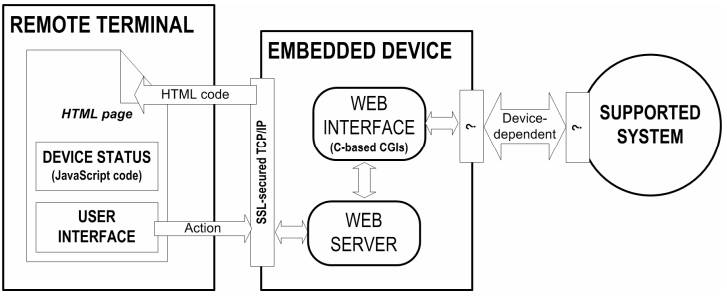


Fig. 3. Web software logic as the interface between the browser and the supported system

Regarding application efficiency, all software components have been developed using a thread-based architecture when concurrent actions (e.g. serving network petitions, etc…) are needed. This improves performance thanks to the parallelization provided by the thread-based approach.

However, special care must be taken with the memory threads use, as they may exhaust the limited memory resources available in the AXIS platform. Memory depletion issues cause erratic behavior not only the consumer applications, but also in other programs and the operating system.

Limiting the number of threads an application can launch is a good mechanism to prevent memory-starvation issues. It may reduce functionality, but it avoids more severe consequences than those software malfunctions can cause over trusted systems; such failures might be exploited by malicious people.

3.3 Real-Life Application: "Indalo", a "Premier" Alarm Panel Controller

All this theory has been brought into reality in the form of a device implementing a web-based interface that allows administering a "Premier" alarm central device. The alarm central, in its turn, is connected to several alarms distributed along a corporation. This real-life appliance has been successfully deployed into the local network of a national aero-spatial company, allowing security staff to detect unauthorized accesses to its different secured rooms.

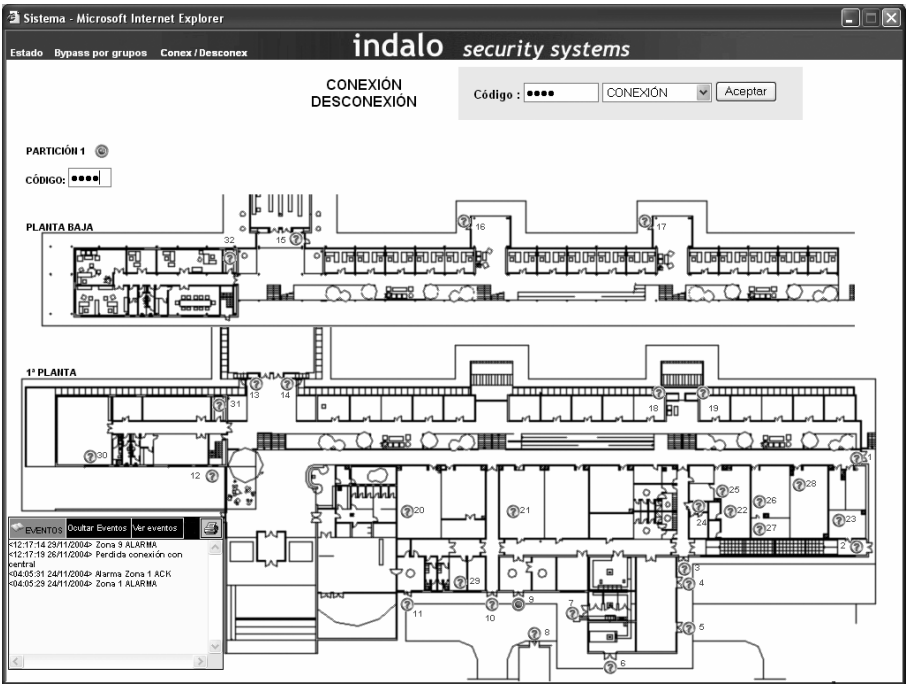


Fig. 4. Screenshot from web interface of "Indalo" access control management device

The "Indalo" device – name of this customization for this general-purpose embedded system - can be accessed through the corporation network, allowing users to have the correct administration rights, supervise and administer the whole access-control system. Both on demand and "real-time" HTML content are provided by the box, being the later periodically updated so it always displays an up-to-date status of the system.

In addition, system's logic provides additional administration web-pages, allowing users to create customized statistical reports based on different policies, such descriptions about non-allowed accesses or other system events that have taken place in a given lapse of time.

As commented above, we are forced to use CGI executables for both transmitting actions taken on web pages to the alarm central and reflecting the status of the different alarms administered by the alarm central.

CGI executables that analyze the status data received from the alarm system are excuted periodically, feeding the HTML pages by creating the JavaScript code so the browser can show the status of the different alarms (bypassed, alarmed, etc.).

Commands to be taken over the alarm system are forwarded from the browser to other CGIs, which analyze the command type and its parameters and send the correct command packet structure to the alarm system via a RS-232 serial connection.

Result of commands operation received from the alarm system is then parsed and JavaScript code is created so HTML pages can reflect the result.

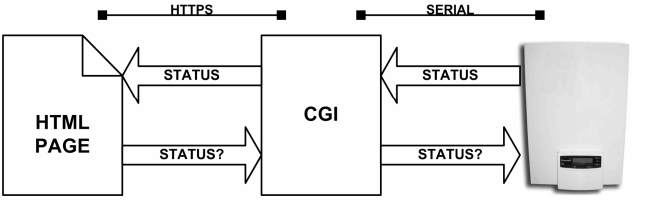


Fig. 5. Embedded system using CGIs to communicate with "Premier" alarm control panel

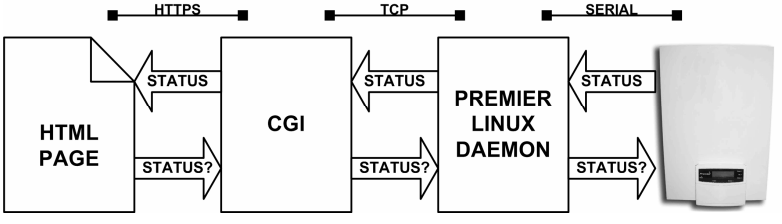


Fig. 6. Separating device-dependent code to be on daemons and HTML-dependent code on CGIs

With the objective of having a more flexible system where CGIs need not be modified when so are low-level components, we have decoupled low-level libraries that communicate with external hardware from CGIs.

Within this new structure, libraries are now located into new executable components designed to work as Linux daemons that serve to requests coming from CGIs using local network sockets. The combined use of sockets and threads allows for multiple CGIs petitions to be simultaneously served.

There is an inherent trouble derived from the way Linux O.S. handles serial ports, which means that concurrent accesses to serial port may cause data sent from different CGIs to be interleaved, and thus making the "Premier" alarm panel receive mixed data it cannot understand.

We have solved this drawback using a shared queue where the daemon inserts all petitions coming from CGIs, so they are served in a serialized manner.

4 Conclusions

The Linux-based embedded system described in this paper provides a flexible and expandable platform that can be used for different purposes.

Once connected to the corporative network, the system can be remotely managed using web connections, so any authorized user located anywhere can make use of its features with a PC, cellular phone or any other device supporting web-surfing.

Security is another strong point in this platform. All information received and sent by it is encrypted using SSL technology. Also, users accessing the device will need to identify themselves; this prevents users from using features other than the ones they are allowed to.

The functionality of this system can be remotely extended, or transformed to support new applications, with just a new firmware (software) update.

This provides a more cost-effective solution for both final users and developers than what was offered by previous systems, where changes in the system meant that the replacement of hardware parts, or displacement of technical staff was necessary.

In a world where companies are increasingly giving services to international customers, the capability of providing distant support makes all the difference.

The development of this kind of system shows that –in spite of the restrictions in memory and CPU resources- it is possible to build a powerful general-purpose embedded device, which can be programmed to serve a specific purpose.

This goal has been achieved thanks to a design where efficient software technologies and techniques, as well as common hardware interfaces, are the keys to success.

Acknowledgements. The work described in this paper has been funded by the Ministerio de Ciencia y Tecnología within the I+D+I National Program through the project with reference number TEC2006-08430.

We´d also like to thank ISIS Engineering (Seville) for providing us with prototypes, and Medina-Garvey electrical company for letting us use their facilities.

References

1. Konicek, J., Little, K.: Security, ID Systems and Locks = The Book on Electronic Access Control. Butterworth-Heinemann (1997)
2. Yaghmour, K.: Building Embedded Linux Systems. O'Reilly, Sebastopol (2003)
3. Boyer, S.A.: SCADA = Supervisory Control And Data Acquisition, 2nd edn. SA – The Instrumentations, Systems and Automatic Society, New York (1999)
4. Information technology – Local area networks – Part 3: Carrier sense multiple access with collision detection. IEEE 802.3 (1993)
5. Ibarra-Manzano, M.A., Almaza-Ojeda, D.L., Aviles-Ferrera, J.J., Avina-Cervantes, J.G.: Access Control System Using an Embedded System and Radio Frquency Identification Technology. In: IEEE, Electronics, Robotics and Automotive Mechanics Conference – CERMA 2008, pp. 127–132. IEEE Press, Los Alamitos (2008)
6. Rescorla, E.: HTTPS = HTTP over TLS. IETF RFC 2818 (2000)
7. Postel, J., Reynolds, J: Telnet protocol specification. IETF RFC 854 (1983)
8. Yan-ling, X., Wei, P., Xin-guo, Z.: Design and implementation of secure embedded systems based on Trustzone. In: International Conference on Embedded Software and Systems – ICESS 2008, Sichuan, pp. 136–141. IEEE Press, Los Alamitos (2008)
9. University of Southern California: IP = Internet Protocol. IETF RFC 791 (1981)
10. Leon-García, A., Widjaja, I.: Communication Networks, 2nd edn. McGraw-Hill, New York (2003)
11. Technical guide of USB 2.0. USB Implementers Forum (2001), `http://www.usb.org`
12. GPRS – Service Description; Stage 2. ETSI GSM 03.60 (2000)
13. Siever, E., Figgins, S., Weber, A.: Linux in a Nutshell, 4th edn. O'Reilly, Sebastopol (2003)
14. Hypertext Processor Scripting Language – 5.0.2. The PHP Group (2004), `http://www.php.net`
15. Mitchell, S.: Designing Active Server Pages. O'Reilly, Sebastopol (2000)
16. Robinson, D., Coar, K.: CGI = The WWW Common Gateway Interface Version 1.1. IETF RFC 3875 (2004)

Recognizing Customers' Mood in 3D Shopping Malls Based on the Trajectories of Their Avatars

Anton Bogdanovych[1], Mathias Bauer[2], and Simeon Simoff[1]

[1] School of Computing and Mathematics, University of Western Sydney, Australia
{A.Bogdanovych, S.Simoff}@uws.edu.au
[2] Mineway GmbH, Science Park 2, Stuhlsatzenhausweg, 66123 Saarbruecken, Germany
mbauer@mineway.de

Abstract. This paper proposes a method to assess the cognitive state of a human embodied as an avatar inside a 3-dimensional virtual shop. In order to do so we analyze the trajectories of the avatar movements to classify them against the set of predefined prototypes. To perform the classification we use the trajectory comparison algorithm based on the combination of the Levenshtein Distance and the Euclidean Distance. The proposed method is applied in a distributed manner to solving the problem of making autonomous assistants in virtual stores recognize the intentions of the customers.

Keywords: 3D virtual worlds, Trajectory recognition, Avatar, e-Commerce.

1 Introduction

Starting as game-oriented technology 3D Virtual Worlds became one of the few successful online businesses that are making money on the Web [1]. The popularity of Virtual Worlds grows and the demand for them to be applied to a wider range of domains (i.e. Electronic Commerce, Tourism, Museums) becomes more and more explicit. Another trend associated with the growth of Virtual Worlds is the demand for mixed societies, which can be populated by both humans and autonomous computational agents. This demand is stimulated by the desire of businesses establishing in Virtual Worlds to provide customers with adequate assistance and, at the same time, save on human resources by employing autonomous agents.

A particularly interesting case of E-Commerce environments that would benefit from the presence of autonomous shopping assistants represent 3D Shopping Malls. 3D Shopping Malls are online stores located in various Virtual Worlds (e.g. Second Life). Due to the fact that all the participants in 3D Virtual Worlds share similar embodiment and the environment allows for full observation of customer actions – 3D Virtual Worlds offer a potentially better platform for development of intelligent shopping assistants than form based interfaces [2].

In 3D Virtual Worlds the range of possible actions is much wider. Every mouse or keyboard event can be associated with a number of attributes that play a role in triggering this event, and those attributes can be easily analyzed and extracted from the environment. Every movement of an avatar[1] can be precisely described by a set of

[1] Avatars are graphical representations of humanoids in Virtual Worlds.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 745–757, 2009.
© Springer-Verlag Berlin Heidelberg 2009

transformation vectors and each of those vectors can be easily related to the surrounding objects, helping to build a mathematical model of the training data. Moreover, analyzing the movements of the human can reveal information about his/her cognitive state [3], which is difficult to figure out in form-based interfaces.

On the example of museum visitors [4] it was shown the choice of the correct assistance strategy required by an individual is highly dependent on his/her cognitive state. Some aspects of the cognitive state can be directly recognized from the trajectory of movement of this individual. In [3] it is described how motion-based information can play an important role in recognizing various aspects of the cognitive state of the user including a degree of user's commitment to a goal and the goal itself. The paper suggests using the standard CAPRI algorithm for mining the association rules from the movement profile of the users, which in our case may be used for selecting an appropriate assistance style, and determining the relevance of a particular piece of information the assistant is about to present to the user. Another idea proposed in [3] is collecting a number of motion sequences in a certain area and then determining clusters of similar motions, where each cluster corresponds to a specific value of the cognitive state.

This paper extends the work presented in [3] and applies it to the domain of 3D Shopping Malls. Here we provide a details of the clustering algorithm, outline the method for calculating the distance between motion sequences and conduct a set of experiments for validating this method. It is also shown how this clustering relates to recognizing the cognitive state of the customers visiting virtual shops. The suggested clustering method is based on the Levenshtein Distance and the Euclidean Distance measures, so it can capture the geometrical features of the motion sequences. While the same problem could have been solved using classical data mining methods, using geometrical features is quite important for the domain of 3D E-Commerce as such information can be used for relating the motion sequence to a particular location (product).

Another contribution of our work is "Distributed User Modeling" that is a technique for introducing the distributed data mining into the domain of 3D E-Commerce.

The remainder of the paper is structured as follows. Section 2 explains the concept of cognitive state, discusses how it can be assessed on the basis of a trajectory and outlines the testing scenario. Section 3 presents our distributed user modeling approach for solving the problem with analyzing the trajectory of the avatars not directly controlled by the assistant agent. In Section 4 we outline the algorithm that we use for discovering some aspects of the customer's cognitive state on the basis of his/her avatar's trajectory. Section 5 demonstrates the results of the experiments we have conducted. Finally, Section 6 presents some concluding remarks and the direction of future work.

2 Assessing Cognitive State

Cognitive state is broad term used in different disciplines. Below is the definition that most accurately reflects what is understood by the cognitive state in this paper.

Definition. ***Cognitive State*** *is the state of a person's cognitive processes*[2]*. In DAI*[3] *cognitive state is usually associated with intentions, beliefs and desires of an individual* [5].

[2] http://www.dictionary.com

[3] Distributed Artificial Intelligence.

It is not the goal of this paper to analyze different aspects of the cognitive state and provide a comprehensive study on how they can be learned. Instead, our goal is to show the potential that 3D E-Commerce domain provides in analyzing it. Therefore, the presentation here is limited to analyzing only one aspect of the cognitive state, namely the mood of the customer inside a virtual shop.

The mood aspect of the cognitive state has received particular attention in ethnography [4]. Based on the mood of the people visiting art expositions in museums researchers identified four distinct categories of visitors, briefly summarized in [4].

1. The *ant visitor* spends a long time to observe all exhibits, stops frequently and usually moves close to walls and exhibits, avoiding empty spaces.
2. The *fish visitor* moves preferably in the center of the room, walking through empty spaces. Fish visitors do not look at details of exhibits and make just a few or no stops; most of the exhibits are seen but for a short time.
3. The *grasshopper visitor* only sees the exhibits which comply with grasshopper's interests. These personal interests and pre-existing knowledge about the contents of the exhibition guide the grasshopper. The grasshopper quickly crosses empty spaces, but the time spent on observing selected exhibits is quite long.
4. The *butterfly visitor* frequently changes the direction of visit, usually avoiding empty spaces. The butterfly sees almost all the exhibits, stopping frequently, but times vary for each exhibit.

The problem of assisting a customer in a virtual shop is very similar to the problem of assisting a visitor of an art exhibition in a museum. In fact, most of the existing virtual shops in the domain of 3D E-Commerce have a similar set up as an art exhibition. Many of such shops, due to the high price of 3D modeling, have the pictures of the products located alongside the walls of the store.

2.1 The Poster Shop Scenario

In our experiments we have used a poster shop, where the goods offered for sale were various graffiti posters placed on the walls of the virtual room.

Due to a close similarity with the art exhibition domain we decided to use the same 4 mood state values: "Ant", "Grasshopper", "Butterfly" and "Fish" to represent Buyer's behavior in the shop room. However, in contrast to the art exhibition in our scenario only one room was used for conducting the experiments. Another difference with the art gallery scenario from [4] is that we need to analyze the trajectories of the visitors dynamically, meaning that it is not acceptable to wait until a participant exits the room to be able to classify their mood. In our case mood classification is required to be completed before the participant approaches the assistant. Having these limitations requires a slight modification of the legends behind the behaviors presented in [4].

As adapted to our scenario, in the "Ant" state the main task of the user is considered to be walking around the room and absorbing the visual information presented there. In the "Grasshopper" state a user is focused on particular items in the room (graffiti posters) and requires more information about them. In the "Butterfly" state the user is experiencing the problem with either navigating the Virtual World or with the presented information. Being in "Fish" state means that the user is focused on some task outside

the shop area, wishes to pass through the room as quickly as possible and doesn't want to be distracted from the main activity.

The mood of the visitors of the graffiti poster shop affects the way they perceive information. For the assistance purposes knowing it is very important to determine which strategy to select and to understand whether any assistance is required at all. Clearly the person in the "Fish" state would not be very excited about hearing the information about graffiti posters and the person in "Butterfly" state experiencing navigational problems would first like to know how to solve the problem and only then may express some interest in the poster exhibition.

3 Distributed User Modelling

In the above presented scenario the Assistant agent has to be able to asses the cognitive state of the Buyer approaching the Assistant to select an appropriate assistance strategy. As shown above, the trajectory of the Buyer could reveal some of the aspects of the cognitive state. Analyzing a trajectory of another avatar, however, is a very challenging task associated with a number of problems. The Assistant agent must constantly observe every Buyer that enters the room and translate the movements of the Buyers into arrays of landmarks. Without being able to directly acquire this information from the system, analyzing the cognitive state becomes nearly impossible.

Obtaining landmarks and their precise coordinates just from observation of the movements of one avatar through the field of view of another avatar is a very challenging task. This task could be simplified by letting the system provide every agent with detailed information about all the other agents (including the positions of the corresponding avatars at a given time). Such simplification, however, increases computational load and raises privacy concerns.

To be able to make the trajectory recognition achievable and, at the same time, ensure the privacy of participants we propose the following decentralized solution to the problem. Each autonomous agent only observes its principal and dynamically updates the user model of the principal (in our particular case the user model only constitutes the cognitive state). When some other agent (i.e. the Assistant) needs to obtain some information from the user profile of this agent (Buyer), instead of trying to observe the behavior of its avatar and use sophisticated modeling techniques, it simply sends a direct request to the agent "responsible" for the corresponding avatar. If the other agent agrees to share the information it will reply with the relevant part of its user profile.

Such decentralized solution is feasible because the duality (agent/principal) is a general feature of Virtual Institutions technology employed for the development of our scenario. In Virtual Institutions [2] every participant is integrated into the system via such architecture. The proposed approach can significantly reduce the amount of computations and the size of the stored data. It also permits easier control over privacy (e.g. if a participant doesn't want to be observed he/she just prohibits the agent to share personal information with others or may even select which aspects of the user profile can be shared and which aspects are private). The decentralized approach also helps to easily use the characteristics of the user profiles of the surrounding agents in the system as the attributes for other kinds of machine learning. The Virtual Institutions architecture

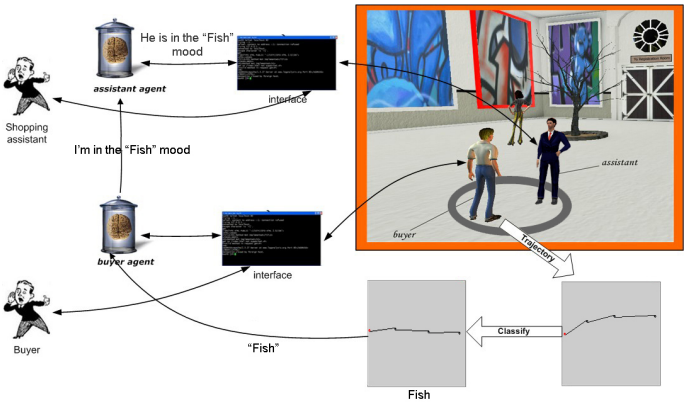


Fig. 1. Distributed User Modeling

supports that an agent that observes the behavior of its principal may directly communicate with the agents attached to the avatars currently visible to it and ask them to share a particular part of their profile. The elements of the obtained user profile can then be used as the input for a classifier in implicit training of the assistant agent [6].

Figure 1 graphically presents the idea of distributed user modeling. It outlines the case of a human controlling the avatar marked as "Buyer" and an autonomous agent controls the avatar of the "Assistant". In the beginning of the scenario outlined in the figure the Assistant representative agent notices a Buyer approaching its avatar. To be able for the Assistant to detect that it was approached the notion of *audibility distance* is used. The audibility distance represents the radius of the imaginary circle surrounding the avatar (audibility zone), which is used to determine the range in which everything that is said by any other avatar will be heard, while outside of this range nothing can be heard. Audibility distance is a very useful concept for facilitating social interactions and providing a natural way to filter the communications. Another purpose of the audibility zone in our system is that the fact of entering the audibility zone of one avatar by another avatar means that the first avatar was approached by the second one.

Once approached, the Assistant requests the description of the Buyer's cognitive state from the Buyer representative agent. In its current form the state corresponds to the label describing the mood of the human. This label is acquired by the Buyer representative agent through comparing the current approaching trajectory of the Buyer with the set of predefined prototypes and extracting the label of the prototype that has the closest match. In the scenario outlined in the figure this label is "Fish".

Once the label is assigned it is sent back to the Assistant representative agent. Before starting a conversation, the Assistant already knows that it shouldn't bother the visitor with additional information about the posters and if asked directly, its responses have to be very short and precise.

Notice that it is also possible that the Assistant avatar is controlled by the human. In this case the corresponding autonomous agent will inform the human about the mood of the approaching participants and the human will be able to use this information for selecting the right strategy, and will train the agent accordingly.

4 Trajectory Comparison

The cognitive state of the Buyer is estimated on the basis of its approaching trajectory. In this paper we do not intend to prove the connection between the movement of the avatars and the cognitive state of the humans generating these movements, but rely on the outcomes of the research presented in [4] to make this link. We use the adapted versions of the four trajectories presented in [4] as the basis for the trajectory comparison.

Each of the four trajectory prototypes is stored in the classification list. In order to classify the approaching trajectory of an avatar we compare it with every trajectory in the classification list and identify the most similar one from the list. The label associated with the resulting trajectory is considered to be the result of the classification.

Technically, the trajectories are specified as arrays of landmarks. Each of the landmarks corresponds to a position of an avatar in a given moment. The position is permanently updated by the system every 50 Ms, so the information about avatar's velocity is easily reconstructed from the distance between two neighboring landmarks. This simple representation allows efficient trajectory classification. To increase the performance of the classification on the first step of the algorithm the irrelevant landmarks (noise) are removed using the approach presented in [7].

After this, a combination of Levenshtein Distance and Euclidean Distance algorithms is applied to compare the analyzed trajectory with each trajectory stored in the classification list. As the result of the comparison, the trajectory from the classification list with the lowest distance value is selected and the corresponding text value is extracted to be used as a behavior label for the cognitive state of the human.

4.1 Levenshtein Distance

The Levenshtein Distance is the algorithm normally used to measure the distance between two strings. It determines the minimum number of operations needed to transform one string into another given string, where possible operations are insertion, deletion, or substitution of a single character [8]. The steps of the Levenshtein Distance Algorithm [8] are presented in Table 1. Here s and t are the two strings being compared, n – the length of string s and m – the length of string t.

In order to apply the Levenshtein Distance to the trajectory comparison we propose the following modifications to the original algorithm. Firstly, instead of comparing the Strings we compare arrays of landmarks. So each of the $s[i]$ and $t[j]$ will be a point in a 3-dimensional system of coordinates – (x_i, y_i, z_i) and (x_j, y_j, z_j), respectively.

Second change is the replacement of the *cost* assessment model. In the original algorithm the cost can be seen as the actual distance between two characters. This cost model is very simple and is equal to "0" if the two characters are similar and is equal to "1" if the characters are different.

In our case we are dealing with arrays of landmarks instead of characters. Each landmark has a unique coordinate in the 3-dimensional space and, therefore, instead of just having "0" and "1" we can employ a more appropriate distance measurement technique, namely, Euclidean Distance.

The Euclidean Distance between two points in a 3D space is calculated as follows:

$$D_{euclid} = \sqrt{(p_1.x_1 - p_2.x_2)^2 + (p_1.y_1 - p_2.y_2)^2 \times (p_1.z_1 - p_2.z_2)^2} \qquad (1)$$

Table 1. The Steps of the Levenshtein Distance Algorithm

Step	Description
1	Set n to be the length of s (first string). Set m to be the length of t (second string). If n = 0, return m and exit. If m = 0, return n and exit. Construct a matrix containing 0..m rows and 0..n columns.
2	Initialize the first row to 0..n. Initialize the first column to 0..m.
3	Examine each character of s (i from 1 to n).
4	Examine each character of t (j from 1 to m).
5	If s[i] equals t[j], the cost is 0. If s[i] doesn't equal t[j], the cost is 1.
6	Set cell d[i,j] of the matrix equal to the minimum of: a. The cell immediately above plus 1: d[i-1,j] + 1. b. The cell immediately to the left plus 1: d[i,j-1] + 1. c. The cell diagonally above and to the left plus the cost: d[i-1,j-1] + cost.
7	After the iteration steps (3, 4, 5, 6) are complete, the distance is found in cell d[n,m].

Here D_{euclid} is the Euclidean distance, p_1 and p_2 – the landmarks for which the distance is measured and (x_1, y_1, z_1) and (x_2, y_2, z_2) – coordinates of p_1 and p_2.

In theory the values of the Euclidean Distance can vary between "0" and infinity, practically the distance is always limited by the dimensions of the space where the measurement is taking place. The cost value in the original Levenshtein Distance algorithm is required to be normalized (take values in the [0,1] interval). Therefore, instead of using pure distance value we use the following equation:

$$cost = \frac{D_{euclid}}{\sqrt{R.width^2 + R.height^2 + R.depth^2}} \tag{2}$$

Here $cost$ is the value that should be used instead of "1" on the step 5 of the Levenshtein Distance algorithm. And $R.width$, $R.height$, $R.depth$ are the dimensions of the room used for trajectory classification. In case the trajectory comparison is required to be done outside any of the rooms, R should correspond to the space where the experiment is conducted with $width, height, depth$ being the dimensions of this space.

5 Experiments

To verify the proposed method for trajectory recognition and classification of the cognitive state as well as to test its accuracy we have conducted a series of experiments.

The aim with the assessment of the cognitive state was not to predict the actual cognitive state of the user, e.g. whether a user really was expressing fish browsing style or was rather a grasshopper. Instead, we wanted to prove that the trajectory recognition method based on the combination of Levenshtein Distance and Euclidean Distance is an appropriate trajectory clustering technique.

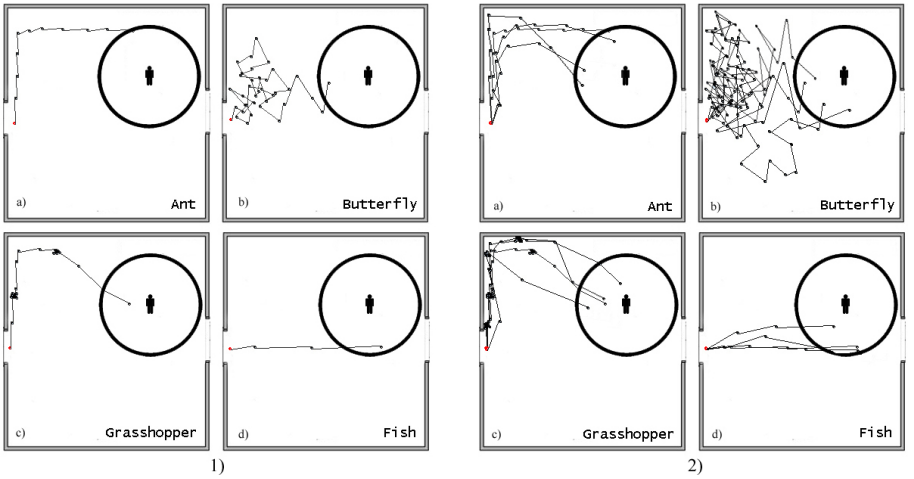


Fig. 2. Trajectories Used for Training and Experiments

5.1 Design of Experiments

For our experiments we implemented the scenario outlined in Figure 1. Test subjects playing the buyer role entered the poster shop room and the assistant had to identify their mood. The Virtual World consisted of the 3 rooms, only one of which was used for our experiments. The schematic representation of the room is shown in Figure 1.

To asses the mood of the buyer each individual agent was observing the trajectory of its principal and comparing the current trajectory with each of those in the classification list. The classification list contained 4 prototypes as displayed in Figure 2. 1.

Here each trajectory is shown as a set of landmarks connected by lines. Each of the landmarks corresponds to the position of the avatar at the moment of measurement. Each new measurement was conducted every 50 Ms. The schematic representation of the poster shop is added to show the context under which the trajectories were obtained. Each of the landmarks is projected onto the corresponding position in the shop room and the label describing the trajectory is shown in the bottom right corner of the room. The figure consists of 4 duplicates of the poster shop marked as “a)”, “b)”, “c)” and “d)”. The trajectories on each of those duplicates corresponds to one class of the cognitive state as marked in the picture. The black solid figure present in each copy of the poster shop room represents the autonomous agent associated with the assistant and the circle around it represents the audibility zone.

The trajectory in Figure 2.1 a) corresponds to the case when a buyer enters the shop not aware of its content. Once inside the room the buyer moves along the wall with a moderate speed checking out the presented posters. Here we wanted to present the case of a curious browser, who has no specific interests or knowledge about the presented products and wants to make a sound decision by browsing through all the posters presented there. This trajectory was associated with the label “Ant”. This trajectory is characterized by the monotonic speed of the user movement along the walls.

In Figure 2.1. b) we present the case of the participant randomly walking around expressing a high degree of confusion. Such a trajectory is typically generated by novice

participants who are not yet quite familiar with controlling their avatars. Label "Butterfly" is assigned to this trajectory. The key characteristic of this movement class is that a participant frequently changes the movement direction and returns to the location close to initial point a number of times.

Figure 2.1 c) illustrates the case of the visitor of the room who has particular interest in some posters and no interest in the others. From the distance between landmarks it is clear that the browsing speed is not constant. In front of two pictures the human stopped for a while and then headed very fast straight towards the exit. This trajectory is labeled as "Grasshopper". On the picture the groups of landmarks placed closely together do not represent very short movements but are due to the fact that during recording a landmark is added after a constant interval of time even if there was no movement produced.

Finally, Figure 2.1 d) illustrates the case when a participant has no interest in buying any posters and shortly after entering the room quickly walks towards the exit (using the poster shop as a corridor) in order to continue with some activities in the next room. The label we use for this behavior is "Fish". The key characteristics here: the high speed of movement, which can be recognized from the distance between the landmarks and the fact that the participants moves in the middle of the rooms, away from its walls.

As it is clearly seen on the pictures, each of the trajectories was recorded from the moment a human entered the room until the moment the corresponding avatar approached the assistant within the audibility distance. In our experiments once the avatar entered the audibility zone of the assistant agent the recording of the trajectory was finished, the currently recorded sequence of landmarks was compared with each of the prototypical 4 sequences and the trajectory with the lowest Levenshtein Distance was selected as the class describing the cognitive state of the human. Then, the corresponding label associated with this trajectory was sent by the autonomous agent of the buyer to the autonomous agent of the Assistant to inform it about the mood of the buyer. This information was further used by the assistant to decide whether to offer help (the help should not be offered to "Fish") and what kind of assistance is required (i.e. the "Butterfly" participant needs a different type of assistance than "Ant" or "Grasshopper").

To validate the trajectory recognition method we conducted a series of experiments with the set of 50 different movement sequences executed in the poster shop. The human operator playing the "Buyer" role was told about the distinct characteristics of each of the 4 classes presented earlier and then was asked to produce 10 different movement patterns for each of the classes so that those patterns would match the given descriptions and at the same time would be distinct. Each of the 10 patterns would end with buyer approaching the assistant. The result of the classification of buyer's trajectory by the assistant agent was printed in the chat window.

Table 2 outlines the results of the conducted experiments. In this table the "**Nr**" column shows the experiment number and the "**Result**" column stores the label printed in the chat window as the result of the classification. The columns marked as "D_{ant}", "$D_{butterfly}$", "D_{fish}", "$D_{grasshopper}$" store the value of the Levenshtein distance between the trajectory from the experiment and the prototypical "Ant", "Butterfly", "Grasshopper" and "Fish" trajectories correspondingly. The "**Correct**" column shows whether the behavior the test subject was intended to demonstrate was correctly recognized.

To give an impression about the movement patterns expressed by the operator Figure 2.2 outlines the results of the 16 out of first 50 experiments we have conducted. We show only 16 (4 per each of the classes) to avoid overcrowding the picture with unnecessary data. The recorded trajectories exemplify the series of movements which begin when the operator driving the avatar of the "Buyer" agent entered the poster shop room and end at the moment this avatar entered the audibility zone of the Assistant agent. For presentation purposes the trajectories are projected onto the schematic representation of the poster shop room. Each of the recordings is classified into one of the four classes as in Figure 2.1. For simplicity of understanding we placed all of the trajectories having the same class into the same part of the figure, which allows for a better comparison.

The 4 trajectories presented in Figure 2.2 a) correspond to the experiments 1–4 in Table 2. In Figure 2.2 b) the experiments 11–14 are outlined. Figure 2.2 c) and Figure 2.2 d) show the experiments 21–24 and 31–34 correspondingly.

One of the goals of the experiments was to highlight the benefits of using geometrical features of the training data for classification. Experiments 41–45 and 46–50 were testing the hypothesis that using the Euclidean Distance as the cost in the comparison will help capturing the specifics of each particular shop and allow for an easy way of expressing the location based preferences. In the experiments 41–45 the test subject was demonstrating the "Ant" behavior, but instead of moving along the upper side of the room was asked to move using similar movement style, but in the bottom part of the room. The same was done for the "Grasshopper" behavior in the experiments 46–50. Using classical data mining methods (which do not take geometrical characteristics into account) for this case would most likely result no difference between moving in the upper part of the room or in its bottom part, both would be classified as being identical. In the 3D E-Commerce domain, however, such a situation is often not acceptable. In particular, for the "Grasshopper" case the posters in front of which users stops make a big difference for making a decision on what kind of information a user might require. Note that for the experiments 41–50 the "**Correct**" column in the results table tells whether the intended behavior ("Ant" and "Grasshopper") is recognized or not.

5.2 Discussion of Results

As shown in Table 2, two of the presented four classes were correctly identified by the system in all cases. These classes are "Ant" and "Fish". Out of 10 experiments per each of those classes 10 were recognized correctly making it 100% classification accuracy.

The "Butterfly" trajectory was also detected very accurately with the precision of 80%, where 8 out of 10 generated examples were recognized correctly. The motion in vertical direction in one of the misclassified trajectories was very low, which became the main reason why this pattern was classified as "Fish". In another misclassified example the operator approached the initial position only twice (while in the prototypical trajectory it happened 4 times) and, therefore, this pattern was also classified as "Fish".

The recognition of the "Grasshopper" trajectory showed the worst precision of only 60%. It proved to be too similar to the "Ant" behavior with all of the 4 misclassified examples being recognized as "Ant".

To gain a detailed understanding for the reasons of misclassification we conducted 50 more experiments with "Grasshopper" and "Butterfly" classes. These experiments

Table 2. Experiments

Nr	D_{ant}	$D_{butterfly}$	D_{fish}	$D_{grasshopper}$	**Result**	**Correct**
1	0.46	3.58	2.03	0.70	ant	y
2	0.45	3.13	2.02	0.96	ant	y
3	0.63	3.13	2.02	0.96	ant	y
4	0.50	3.26	1.62	1.07	ant	y
5	0.68	3.07	2.60	1.42	ant	y
6	0.47	3.15	2.05	1.07	ant	y
7	0.56	3.02	1.87	0.82	ant	y
8	0.65	2.81	1.86	1.41	ant	y
9	0.55	2.83	1.97	0.96	ant	y
10	0.80	3.16	2.79	1.29	ant	y
11	6.05	3.80	3.58	6.22	fish	**n**
12	5.94	2.99	5.11	5.58	butterfly	y
13	3.23	2.57	2.39	3.85	fish	**n**
14	7.04	3.90	5.86	6.03	butterfly	y
15	4.41	2.69	3.65	4.44	butterfly	y
16	4.92	3.19	4.04	5.09	butterfly	y
17	4.54	2.58	3.95	4.46	butterfly	y
18	6.84	3.27	4.36	6.03	butterfly	y
19	6.37	3.28	5.51	5.73	butterfly	y
20	5.66	3.06	6.77	5.10	butterfly	y
21	0.67	3.99	4.82	1.27	ant	**n**
22	1.05	6.67	4.82	0.91	grasshopper	y
23	0.94	3.66	5.01	0.82	grasshopper	y
24	1.36	3.99	6.04	1.34	grasshopper	y
25	1.48	4.02	6.57	1.45	grasshopper	y
26	1.21	4.21	5.98	1.26	ant	**n**
27	1.27	4.17	6.71	1.21	grasshopper	y
28	1.05	4.03	5.68	1.11	ant	**n**
29	1.44	4.96	8.04	1.22	grasshopper	y
30	0.98	4.35	7.26	1.14	ant	**n**
31	2.46	3.37	0.34	4.72	fish	y
32	2.98	5.12	0.50	5.98	fish	y
33	2.89	4.85	0.38	5.89	fish	y
34	3.42	6.73	0.70	7.02	fish	y
35	2.56	3.95	0.36	5.12	fish	y
36	2.89	5.28	0.25	6.01	fish	y
37	3.04	4.54	0.62	5.81	fish	y
38	2.79	4.98	0.26	5.69	fish	y
39	2.72	4.08	0.35	5.47	fish	y
40	3.69	7.53	0.79	7.56	fish	y
41	5.24	8.05	2.64	8.52	fish	**n**
42	5.83	7.91	3.30	8.73	fish	**n**
43	5.52	7.20	2.82	8.28	fish	**n**
44	6.83	8.25	3.98	9.34	fish	**n**
45	8.07	9.32	4.94	10.44	fish	**n**
46	9.86	11.17	7.37	12.50	fish	**n**
47	8.51	9.84	6.07	11.21	fish	**n**
48	8.71	10.07	6.23	11.36	fish	**n**
49	7.25	8.70	4.83	9.93	fish	**n**
50	6.30	7.93	3.87	8.98	fish	**n**

showed that for the "Grasshopper" class the irregularities in speed (expressed by different distance between landmarks in "Grasshopper" class) had higher importance for the classification than the fact of a participant stopping. The trajectories with similar number of stops as in the "Grasshopper" class but with relatively constant movement speed in between stops had a very high risk of being classified as "Ant", while movement patterns with clear speed irregularities were more likely to be classified as "Grasshopper".

For the "Butterfly" class our initial hypothesis that the amplitude of the vertical movement as well as the number of returns to the initial position are the key factors in misclassification proved to be correct. The misclassified trajectories had the lowest value of the Levenshtein distance when compared with the trajectory from the "Fish" class, as the generated examples were too distinct, so the example with the fewest number of landmarks had the lowest distance value.

Confirming our original hypothesis, none of the experiments with numbers 41–50 have resulted in "Correct" classification. The use of geometrical features of the trajectories resulted in the classification being very sensitive to the actual positions of the posters. This indicates that our method can potentially distinguish between items located in different positions even if the same movement style is used to approach them.

6 Conclusions

In this paper we have showed how the trajectory of avatar's movement can be used to assess some aspects of customer's cognitive state. The selected method has proved to be capable of accurate trajectory classification with only one training example required per each class to be recognized. Furthermore, the use of Euclidean distance measure provides a possibility to make the classification position sensitive, which is a highly desired feature in the domain of 3D E-Commerce. Although, we base our initial assumption that a trajectory is tightly connected with the cognitive state of the human on the existing research and also find this fact intuitively right, in the future we plan to obtain more supporting evidence through additional experiments. Therefore, we are planning to further extend our system and conduct tests where the buyers will be able to evaluate how good the actual shopping mood was recognized, not only the trajectory. We also plan to use the presented method for implicit training of shopping assistants.

References

1. Hunter, D., Lastowka, F.G.: To kill an avatar. Legal Affairs (2003), http://www.legalaffairs.org/issues/July-August-2003/feature_hunter_julaug03.msp
2. Bogdanovych, A.: Virtual Institutions. PhD thesis, University of Technology, Sydney, Australia (2007)
3. Bauer, M., Deru, M.: Motion-Based Adaptation of Information Services for Mobile Users. In: Ardissono, L., Brna, P., Mitrović, A. (eds.) UM 2005. LNCS, vol. 3538, pp. 271–276. Springer, Heidelberg (2005)
4. Chittaro, L., Ieronutti, L.: A visual tool for tracing users behavior in virtual environments. In: AVI 2004: Proceedings of the working conference on Advanced visual interfaces, pp. 40–47. ACM Press, New York (2004)

5. Konolige, K., Pollack, M.E.: A representationalist theory of intention. In: Bajcsy, R. (ed.) Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence (IJCAI 1993), Chambéry, France, pp. 390–395. Morgan Kaufmann publishers Inc., San Mateo (1993)
6. Bogdanovych, A., Simoff, S., Sierra, C., Berger, H.: Implicit Training of Virtual Shopping Assistants in 3D Electronic Instituions. In: Proceedings of e-Commerce 2005 Conference, pp. 50–57 (2005)
7. Perng, C.S., Wang, H., Zhang, S.R., Parker, D.S.: Landmarks: A new model for similarity-based pattern querying in time series databases. In: Proceedings of the 16th International Conference on Data Engineering, pp. 33–42 (2000)
8. Levenshtein, V.: Binary codes capable of correcting deletions, insertions, and reversals. Soviet Physics Doklady 10, 707–710 (1966)

Assembling and Managing Virtual Organizations out of Multi-party Contracts

Evandro Bacarin[1], Edmundo R.M. Madeira[2], and Claudia Medeiros[2]

[1] University of Londrina, DC/CCE, CP 6001, 86051-990, Londrina, PR - Brazil
bacarin@dc.uel.br
http://www2.dc.uel.br/~bacarin
[2] University of Campinas, Computing Institute, CP 6176 13083-970, Campinas, SP - Brasil
edmundo,cmbm@ic.unicamp.br
http://www.ic.unicamp.br/~{edmundo,cmbm}

Abstract. Assembling virtual organizations is a complex process, which can be modeled and managed by means of a multi-party contract. Such a contract must encompass seeking consensus among parties in some issues, while simultaneously allowing for competition in others. Present solutions in contract negotiation are not satisfactory because they do not accommodate such a variety of needs and negotiation protocols. This paper shows our solution to this problem, discussing how our SPICA negotiation protocol can be used to build up virtual organizations. It assesses the effectiveness of our approach and discusses the protocol's implementation.

Keywords: Virtual organization, Multi-party contract, Supply chain, Negotiation, Auction, Ballot, Bargain.

1 Introduction

Virtual Organizations (VO) are dynamic alliances of enterprises that together can take advantage of economies of scale when available [17]. Assembling and managing them is a complex task, due to the many relationships and agreements among their components. One possible way to shape and manage such organizations is via multi-party contracts, which must reflect obligations, rights and interaction modes within a virtual collaboration scenario. They are built by means of some negotiation mechanism.

According to [6], VOs need negotiation protocols that are multi-party and interactive, i.e., the protocol should allow simultaneous negotiation among three or more partners and they should be allowed to refine a received proposal, e.g., by means of a counterproposal.

The process of constructing a VO can be quite complex: the partners should reach a consensus on some issues, whereas there is competition among them on others, and also other issues may demand individual agreement. While existing solutions do not make allowance for these negotiation heterogeneity, our protocol provides mechanisms to develop all those negotiation styles for negotiating a single contract, namely, ballots, auctions and bargains.

The same mechanisms can also be employed individually to build specific marketplaces. For instance, they can be easily configured to provide different auction styles

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 758–769, 2009.
© Springer-Verlag Berlin Heidelberg 2009

(e.g., English and Dutch auctions). This paper highlights such mechanisms by means of showing how they can be set up for a number of distinct marketplaces.

The main contributions of the paper are: (a) it points out that multi-party contracts are a means for describing virtual organizationss; (b) it shows how to integrate three different styles of negotiation (bargain, ballot, auction) to build up a virtual organization using SPICA negotiation protocol; (c) it presents some details of the implementation of the SPICA negotiation protocol.

The paper is organized as follows. Section 2 presents a running example that will be used throughout the paper. Section 3 shows how SPICA negotiation protocol can be used to build different setups of marketplaces. Next, Section 4 describes briefly the protocol's implementation. Then, Section 5 evaluates the approach proposed in the paper. Section 6 presents related work. Finally, Section 7 concludes the paper.

2 The Running Example

The scenario described in this section is used throughout the paper to motivate and exemplify the usage of the SPICA negotiation protocol.

The scenario consists of a number of farms (F_1,F_2,...,F_n), a few orange processing companies (PC_1,PC_2,...,PC_i) and a railway company (RC). The farms grow orange trees and deliver their crops to the processing companies. A processing company produces concentrated orange juice to be exported. The juice is transported from the company's plant to the nearest harbour by means of the only railway company available in that region.

There is a standard contract model that processing companies use to buy oranges. The main contract's provisions are shown in Figure 1.[1] Note that *pj*, *ff* and *pf* are the model's parameters.

(1) The price paid by the PC will cover the production costs plus a percentage of the juice value (*pj*) in the commodities market.
(2) The harvest and transportation from farm to industry is done at the PC's expense.
(3) If the supplier is farther than 100km from the processing company, the farm will pay an extra freight fee (*ff*).
(4) If the farm's productivity is below a certain local level, the farm must also pay an extra fee (*pf*).

Fig. 1. Contract model for orange crop

The farms have organized themselves into a Cooperative to better negotiate delivery contracts. The cooperative will choose a processing company and will establish a delivery contract. The chosen company will be the one that proposes the best values for parameters *pj*, *ff* and *pf*. Thus, the farms must reach a consensus on what is the best

[1] In fact, these provisions are a simplification of a contract model (*Consecitrus*) Brazilian farmers and orange juice industry have been discussing to be used in negotiation of future crops.

proposal. Finally, the processing company will negotiate the juice transportation with the railway company. They will haggle over the freight fee *rff*.

Notice the following peculiarities in this scenario. There are several competing PCs and the cooperative will choose the best one using an *auction*. However, the farms must agree on this choice via a *ballot*. Since there is just one railway company, the PC is forced to *bargain*. This complex scenario requires a contract that contemplates multi-party negotiation, and bargains, ballots and auctions. As will be seen, we provide a seamless solution that supports these requirements.

3 The SPICA Negotiation Protocol

The negotiation process is guided by a contract template. Negotiators exchange messages that comply with the SPICA negotiation protocol. If there is an agreement, a contract instance is produced. Section 3.1 describes the contract template and the contract instance. Section 3.2 describes the protocol.

3.1 Contract Templates and Contracts

A contract template consists of a set of clauses with blanks to be filled in. Such blanks are referred to by so-called *properties* and the negotiation process aims at assigning values to them. Thus, a contract instance is a contract template with its properties successfully negotiated. The obligations (or rights) stated in a clause may bind (or benefit) several partners.

The contract model depicted in Figure 1 gives rise to a contract template for the scenario presented in Section 2. A contract template is an XML document composed of several sections. One of them contains a set of clauses, the others are used for setting up the negotiation environment. A template's clause respective to the model is presented in Figure 2. It is written in plain English for simplicity.

```
text: The price paid by the @OBLIGED to the
@AUTHORIZED will cover the production costs plus a
percentage worth of #PJ of the juice value in the
commodities market.
depends: (a precondition)
enforces: (a set of regulations)
service: (URLs and other parameters)
authorized: F1,F2,F3,F4,...      obliged: PC
```

Fig. 2. A simplified template's clause

Note that the name of the property to be negotiated (PJ in the figure) is embedded within the clause's *text*. There are a few parameters regarding the clause enactment. There is a precondition (*depends*) that must hold before the processing company should make the payment. In this example, it could be the existence of an formal statement of a trusted company about the juice's price in the market. There is a postcondition (*enforces*) that refers to one or more regulations that must hold after clause enactment.

Regulations state a number of conditions to a product be transfered from a partner to another. In this example, it could be the accomplishment of a few legal procedures established by the local government; in other context, it could be a quality criterion to be meet. *Service* points to a business process to be executed, enacting the clause. *Authorized* lists the parties that will be payed and *obliged* lists the names of the processing companies that will pay the due value. These lists are referred to by the special properties @AUTORIZED and @OBLIGED in the *text*.

3.2 The Protocol

The main data exchanged in a negotiation by means of negotiation messages are *request for proposals* (RFPs), *offers*, *requests for information* (RFIs) and *information* (Info). They convey several parameters that tune up a specific negotiation, identify the sender and the receivers, and help establishing correlation among messages. Only the relevant parameters for the purpose of this paper are presented.

An RFP invites another party to negotiate a set of properties. A negotiator A sends an RFP to a negotiator B asking for a value for one or more properties. More specifically, an RFP conveys three pieces of data: a set of *asked properties*, a set of *assigned properties* and a *restriction*. The example below shows two RFPs. They are written using a simplified notation. The first RFP asks values for properties *ff* and *pp* and imposes a restriction over the value for *ff*: it would only accept values lesser than 3.00, but does not impose any restriction over *pp*. The second one proposes a value for *ff*, asks a value for *pf*, but imposes a restriction over *pf*. The symbols ≪ and ≫ enclose an RFP.

(1) ≪----,{pp,ff},'ff<3.00'≫
(2) ≪{ff:1.96},{pf},'pf<1.15'≫

A negotiator A proposes a value to one or more properties sending an offer to a negotiator B. Negotiator A informs the properties it is interested in and the values it proposes for them. If negotiator B accepts the offer, both negotiators are committed to the proposed values. A negotiator answers an RFP sending back an offer that assigns values to the asked properties. The example below shows offers that answers the previous RFPs. The first offer is a valid answer for RFP (1). It assigns the value 2.3 to property *pp* and the value 2.50 to property *ff*. This value complies with the imposed restriction. The second offer is a valid answer for RFP (2). The proposed value for property *ff* must be exactly the same of the one assigned in the respective RFP. The symbols [and] enclose an offer.

(1) [pp=2.3, ff=2.50]
(2) [pp=0.9, ff=1.96]

An RFI is very similar to an RFP: it asks values for properties, but also lower and upper bounds for them. An Info is similar to an offer: it proposes values for asked properties and also informs upper and lower bounds for them, however, the negotiator which issued an Info is not committed to it.

RFPs and offers are used to build several styles of negotiation that boil down to three basic ones: bargain, ballots, and auctions. Other styles are obtained from different setups of these basic ones. They use a few negotiation messages that are introduced by the

examples that follow. RFIs and Infos do not lead to agreements (they are not committing), but they help constructing better proposals, thus improving the negotiation process.

The scenario presented in Section 2 uses all those styles. Firstly, only one processing company will be chosen. Thus, there is competition among them to decide which one will assign values for properties *pj*, *ff* and *pf*. This is resolved by means of an auction. However, consensus is also needed: the farms must agree on a received bid. This is dealt through a ballot. Finally, the winning processing company has to negotiate property *rff* with the railway company. Since there is only one such a company, a bargain is used.

There are two approaches to organize the negotiation environment. The first one should consist of three separated marketplaces: one runs the auction and selects the winning bid. Then, this bid is submitted to a ballot, verifying if it is accepted by the most of the farms. Finally, the winning (and accepted) processing company will use a bargaining marketplace to negotiate the transportation. The second environment should consist of an integrated marketplace that develops all those negotiations.

Section 3.3 presents the first approach. It shows each negotiation style separately. Section 3.4 shows how the negotiation styles can be integrated in one marketplace.

3.3 Individual Marketplaces

English Auction. This first individual scenario shows the cooperative looking for a processing company which would buy the farms production in a specific season. Recall that the negotiation parameters (i.e., properties) are *pj*, *ff* and *pf* (Section 2). There are several candidate processing companies (PC1,PC2,...). The cooperative (Coop) will choose one of them by means of an auction. It is the auctioneer and it is helped by a notary (Not) which is a trusted third-party. Figure 3 shows the auction running. **(1)** The cooperative asks the notary to collect up to 2 bids within 30s. The auction's subject is described by an RFP. This RFP asks values for properties *pj*, *ff* and *pf*, and imposes a restriction over the proposals for *pj* — only values higher than 1.2 will be accepted. There should be restrictions over the other properties. They were omitted for the sake of simplicity. The notary agrees to run the asked auction step (it might have refused instead). **(2)** The notary broadcasts the auction issue to all bidders. **(3)** The bidders send offers back to the notary. The offers assign values to the asked properties. The notary collects the received bids (i.e., offers) and sends them to the cooperative. The offers are represented by *o1* and *o2* in the figure. **(4)** The cooperative analyses the received bids and comes to conclusion that it could get better bids. Thus, the cooperative builds a new RFP (note the new restriction over *pj*) and asks the notary to develop another auction step. This repeats as many times as wanted. Eventually, no bidder is interested in submitting another bid. Then, **(5)** the cooperative agrees on the best bid of the previous auction step. The fact that the demanded auction step asked for 2 bids does not imply that the auctioneer must agree on 2 bids.

A Dutch auction can be easily run using the same mechanism. However, the auction's subject is described by means of an offer. In step (1), the auctioneer sends an offer to the notary. In (3), the bidders which first agree on that offer win the auction (up to 2 within 30s) . If no bidder agrees on such an offer within 30s, the auctioneer (4) will build another offer and submit it to another auction step. This repeats until there are winning bidders or the auctioneer gives up.

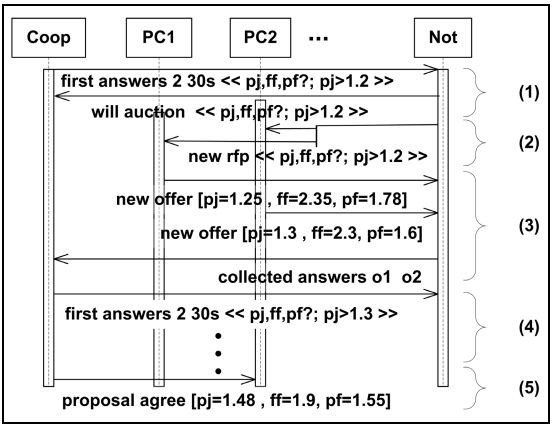


Fig. 3. An auction

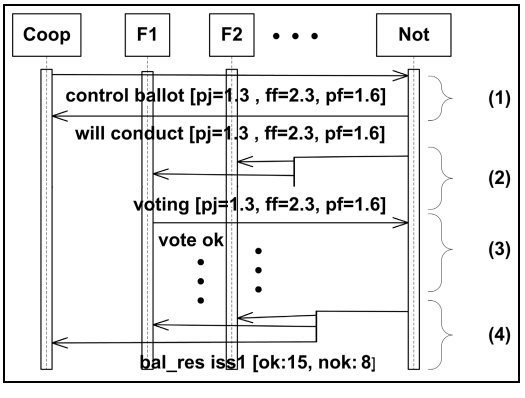


Fig. 4. A ballot

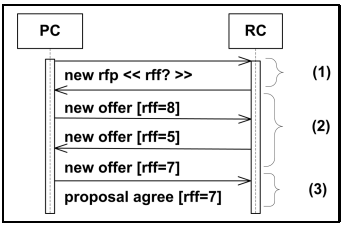


Fig. 5. A bargain

Ballot. The cooperative has found a processing company (the winning bidder in Section 3.3). However, such a choice must be validated by the farms. Now the cooperative runs a ballot helped by the notary (Figure 4). **(1)** The cooperative asks the notary to run a ballot. The ballot issue is described by means of an offer, i.e., the winning bid. The notary accepts conducting the ballot (it might have refused). **(2)** The notary broadcasts the issue to the voters (farms). The cooperative is not a voter in this case. It would be a voter in other setups. **(3)** The farms send their votes to the leader. In this case, they can only agree (vote *ok*) or disagree (vote *nok*). Abstention is also possible. **(4)** The notary collects the votes, counts them and broadcasts the result to all parties (farms and cooperative). In this example, the farms have accepted the winning bid, because more farms have agreed (15) than disagreed (8)

The presented scenario was a "take-it-or-leave-it" ballot. The voters could only have accepted or refused the proposed offer. In step (1), an RFP is used instead of an offer when there are several alternatives for the same property. In this case, the vote will

contain one of these alternatives (instead of *ok* or *nok*). In some setups, a few voters may have veto power. If one of them sends a veto instead of a vote, the ballot is voted down.

Bargain. The processing company (PC) has been chosen and validated by the farms. Now it has to negotiate the transportation with the railway company (RC). They will haggle over a value for property *rff*. This is shown in Figure 5. **(1)** PC asks the freight cost RC would charge for the transportation. **(2)** RC answers that it would charge 8. PC considers is too expensive and makes a counter-offer: 5. RC finds the proposal too cheap and makes another counter-offer: 7. This cycle of counter-offers is repeated as much as needed. **(3)** The process finishes when PC (or RC) reaches a final decision — in this case, PC agrees on the offer.

3.4 Putting Marketplaces Together

The approach presented in Section 3.3 has two main drawbacks. Firstly, the auctioneer has to develop completely an auction and use its own criteria (not the farms') to choose the winning bid. Secondly, it has to submit its chosen bid to a ballot. If the bid is not approved, the auctioneer has to start the auction anew.

In an alternative setup, auction steps are interleaved with ballot sessions: the auctioneer submits all collected bids to the farms at the end of the auction step in successive ballots. If one of the bids is approved, the auctioneer chooses this as the winning bid; otherwise, it runs another auction step.

The role of the auctioneer in this setup can range from neutral to highly interested in a certain outcome. The order that it submits the received bids to ballot may influence the result. Thus, it can rank the bids aiming at efficiency (e.g., submitting first the bids it considers being more probable to be accepted) or according its own interests.

The voters determine the auction step's winning bid. In case no bid is chosen, the auctioneer typically will create a new RFP (or offer), and submit it to a new auction step. The auctioneer can consider the ballot's result of previous steps and try to prepare an RFP that would direct the bids closer the voter's expectations.

4 Implementation in Brief

We have been implementing a framework for the integration of agricultural supply chain (SPICA). The negotiation of contracts is a part of it (SPICA Negotiation Protocol). The core of this negotiation protocol has been implemented. This section presents a few details about such an implementation.

Negotiators and the notary are web services. A negotiator N1 is willing to interact with another negotiator N2 (it might also be the notary). Such an interaction happens by means of an operation being invoked at the appropriate interface.

The framework provides a number of Java classes and Java interfaces to ease the implementation of negotiators. The Web services interfaces described in [2] are directly mapped to Java interfaces. A negotiator should react properly upon receiving a given negotiation message. For instance, whenever a negotiator receives an offer, it should (a)

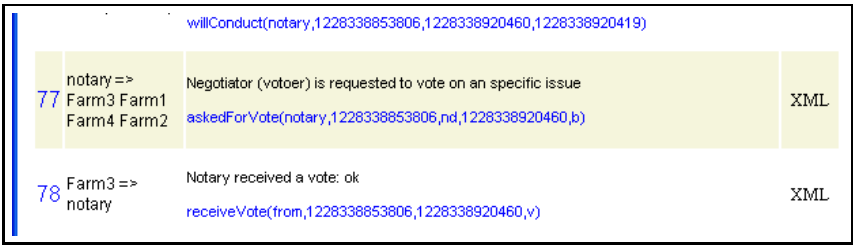


Fig. 6. Logged message

analyze it and decide about agreeing, disagreeing or making a counter-offer; (b) prepare the corresponding answer message, and (c) send it back to the offer originator. There is a default negotiator (DN) that implements this mechanism in a way that a specific negotiator (SN) needs only to override a few methods concerning the decision-making phase (i.e., step a).

There are also classes that help the communication of negotiation messages among negotiators. The framework's design aims at providing specific negotiators the illusion that they are exchanging negotiation messages by means of simple method calls at another local object (i.e., the other negotiator(s)). To do so, the framework provides two classes: *CommunicationAdaptor* and *MessageBroker*. *CommunicationAdaptor* mimics a negotiator. When a specific negotiator (SN1) wants to call a method M at another negotiator (SN2), it calls this method M of the *CommunicationAdaptor*. Then, the *CommunicationAdaptor* serializes the method's parameters (XML-formated) and delivers it to a middleware that transports the message. At the other end, the message reaches a *MessageBroker* which gets the message's parameters and invokes the method M at SN2.

Figures 6 and 7 show an excerpt of messages exchanged among negotiators logged by the system. In Message 77 (Figure 6), the notary asks the negotiators (farms) to vote on a specific issue. The notary uses the *CommunicationAdaptor* to upload this message. In the end, the method *askedForVote* is called at each negotiator. Figure 7 shows an excerpt of the respective message serialized by the *CommunicationAdaptor*.

5 Discussion

This paper shows by means of an example how a multi-party contract can be negotiated using the SPICA negotiation protocol.

Section 3 presented the SPICA protocol. The individual marketplaces approach (Sec. 3.3) was discussed first because most of the negotiation frameworks found in the literature have only one negotiation style, typically auctions or bargains. To the best of our knowledge, none uses a ballot as a foundation of a marketplace. If one would create a marketplace combining an existing auction and a bargain framework, it would be like the one proposed in Section 3.3.

Section 3.4 showed that the same primitives can be combined into an integrated marketplace. Thus, it is possible to explicitly correlate different negotiations. The contract instance produced by such a negotiation naturally shapes a VO. It is noteworthy that

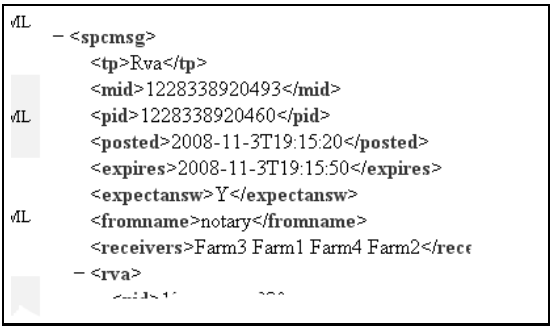


Fig. 7. XML serialized message

VOs are better shaped by means of multi-party contracts than by a set of bi-lateral ones. In this context, negotiation by consensus is quite important but rarely used.

A contract template can be used to describe a VO and its negotiation is the process of building a new VO that endures until the end of the agreed contract. For instance, Section 2 showed a scenario consisting of several actors. It gives rise to two possible approaches. In the first, two unrelated bi-lateral contracts are built: (a) between the processing companies and the cooperative, and (b) between the processing company and the railway. In this setup, the railway company is not aware about its role in the VO. In the second approach, a multi-party contract is used to establish the relationship among farms, processing companies and the railway company. The model presented in Figure 1 should only be enhanced with a clause about property *rff*. This setup allows that the contract provisions focus on a shared goal (i.e., export orange juice and improve profit), shaping a VO where all partners know their role in it.

The SPICA negotiation protocol was designed to provide flexible yet comprehensive negotiation primitives. They are somehow choreographic in the sense that they indicate how the partners should react upon receiving a given message, but does not fully define their expected behaviour. Auctions are a clear example of this. The protocol just defines that the auctioneer will ask the notary to conduct one auction step (not the whole auction). The notary will inform the negotiators about it, collect the bids and send them back to the auctioneer. The auctioneer is the one who is responsible for deciding if there is a winning bid, or if it will try another step, and, even, if it would give up the auction at all. This gives rise to several opportunities. For example, it is possible to run different styles of negotiation without changing the protocol at all. For instance, in Section 3.4 the winning bid (if any) of an auction step is decided by means of a ballot and not by the auctioneer itself. Another example, a segment of the supply chain or a VO may specify different types of behaviour of auctioneers and establish that negotiations within such segment or VO will be done under such specification.

The negotiation primitives build on two basic concepts RFPs and offers. RFIs and Infos may be used for building better proposals to be submitted to a ballot. For instance, a processing company could use RFIs to have a learned guess about the transportation costs and use this knowledge to submit more competitive bids. In this case, it would be a three-level negotiation scenario: (i) a processing company submits RFIs to the

transportation company and (ii) take part of an auction, (iii) and the winning bid is decided by means of a ballot, all simultaneously.

6 Related Work

Our contract model is based on previous work in agricultural supply chains, a very complex kind of VOs [3,2]).

There are several proposals for contract specification. Some of them are designed for specific purposes where the domain of negotiatable items is predefined, like SLAs (e.g.,[7]). This is not our approach. More generic contract specification approaches need an expressive language to describe the commitments agreed among the partners. Several of them use logic-based approaches, like [15,9,8,14,20]. Our approach is different. It is was designed to be used in real agricultural supply chains where the participants are autonomous and heterogeneous. Thus, the process of designing a contract template must be feasible by a team of one TI professional and a lawyer.

A contract expresses commitments among partners. We advocate that a contract in the context of an agricultural supply chain and VOs should express the agreement among more than two partners. However, most of the ones proposed in the literature are bi-lateral, just a few are multi-party, e.g., [20].

Contracts are the outcome of some negotiation process which may be done with some level of software assistance. We proposed automatic negotiation performed by software agents and guided by contract templates [2]. Other proposals also use templates, e.g., [10,4,5]. An alternative for negotiation are matchmaking approaches like [13].

Kallel and others [11] propose a multi-agent negotiation model for a particular contract type of a specific supply chain. The negotiation model consists of a heuristic negotiation protocol and a decision-making model. The authors' approach diverge from ours in two aspects. Firstly, they understand a supply chain as a neighborhood: a focused company, its suppliers and its customers. We consider a supply chain "from farm to fork". To the limit, an alliance (i.e., VO) within a specific supply chain would comprise partners of all supply chain levels. A business process of this wide VO would aim at a complex and long-term event, e.g., the organization of the 2014 World Cup (Brazilian people are not used to eat potatoes in every meal; thus, it would be necessary to increase potato crops timely). Secondly, we propose a generic negotiation protocol rather than a specific one. This prevents us to aim at optimizations (e.g., maximizing profits), but widens the protocol applicability.

Pitt and colleagues [16] propose a voting protocol for multi-agent VOs. This voting protocol characterises the powers, permissions, obligations and sanctions of the voters and is specified by means of Event Calculus. This protocol is used in the context of an agent community. Decisions must be taken during the life-cycle of such a community. The authors' approach is slightly different from ours. They do not focus on establishing an agreement (i.e., contract) for future enactment, but on decisions taken "on-the-fly" during the enactment.

A number of authors use contracts to describe the coordination of activities of partners, e.g., [19,12]. Others use contracts a means of monitoring the fulfillment of the contract's commitments, e.g., [20]. In addition, [1] use a Petri net-base approach to dis-

cuss how a partner should implement its part of the contract complying to the contract's description.

Most of research efforts that combine VOs and contracts are in the context of agent societies, e.g., [18,21]. They use contracts to shape the agents' behaviour, i.e., the actions they might or might not be undertaken regarding to the use of shared resources. They are not business contracts indeed.

7 Conclusions

Contracts can be used to assemble and manage VOs. Multi-party contracts are required in such a context, because a set of bi-lateral contracts can destroy or hide relationships among the partners. Such relationships are hard to model and manage, because they are not homogeneous within a VO. Some issues may demand consensus among the partners. Others, competition among them or exactly two partners must reach a consensus. Thus, it is important that different styles of marketplaces be seamlessly combined in a single coherent assembling process. This paper presented how SPICA negotiation protocol can be used to do so. It also outlined some details of the protocol's implementation.

Future work includes the implementation of an infrastructure for a mechanism to monitor the contract's fulfillment.

Acknowledgements. Research financed by Brazilian Science Foundations CAPES, CNPq, Bio-CORE and FAPESP.

References

1. van der Aalst, W.M.P., Massuthe, P., Stahl, C., Wolf, K.: Multiparty Contracts: Agreeing and Implementing Interorganizational Processes. Technical report, Humboldt-Universität zu Berlin, 2007 (2007) Informatik-Berichte 213
2. Bacarin, E., Madeira, E.R.M., Medeiros, C.B.: Contract e-negotiation in agricultural supply chains. Intl. Journal of Electronic Commerce 12(4), 71–97 (summer 2008)
3. Bacarin, E., Medeiros, C.B., Madeira, E.R.M.: A Collaborative Model for Agricultural Supply Chains. In: Meersman, R., Tari, Z. (eds.) OTM 2004. LNCS, vol. 3290, pp. 319–336. Springer, Heidelberg (2004)
4. Bartolini, C., Preist, C., Jennings, N.R.: A software framework for automated negotiation. In: SELMAS, pp. 213–235 (2004)
5. Chiu, D.K.W., Cheung, S.C., Hung, P.C.K., Chiu, S.Y.Y., Chung, A.K.K.: Developing e-negotiation support with a meta-modeling approach in a web services environment. Decision Support Systems 40(1), 51–69 (2005)
6. Darko-Ampem, S., Katsoufi, M., Giambiagi, P.: Secure negotiation in virtual organizations. In: EDOCW 2006: Proceedings of the 10th IEEE on International Enterprise Distributed Object Computing Conference Workshops, Washington, DC, USA, pp. 48–55. IEEE Computer Society Press, Los Alamitos (2006)
7. Fantinato, M., de Toledo, M.B.F., de Gimenes, I.M.S.: A feature-based approach to electronic contracts. In: CEC/EEE 2006, pp. 34–41. IEEE Computer Society Press, Los Alamitos (2006)

8. Governatori, G., Dumas, M., ter Hofstede, A.H.M., Oaks, P.: A formal approach to protocols and strategies for (legal) negotiation. In: ICAIL, pp. 168–177 (2001)
9. Grosof, B.N., Poon, T.C.: SweetDeal: Representing Agent Contracts with Exceptions Using Semantic Web Rules, Ontologies, and Process Descriptions. Intl. Journal of Electronic Commerce 8(4), 61–97 (2004)
10. Hanson, J.E., Milosevic, Z.: Conversation-oriented protocols for contract negotiations. In: EDOC, pp. 40–49 (2003)
11. Kallel, O., Jaâfar, I.B., Dupont, L., Ghédira, K.: Multi-agent negotiation in a supply chain - case of the wholsale price contract. In: Cordeiro, J., Filipe, J. (eds.) ICEIS (4), pp. 305–314 (2008)
12. Linington, P.F., Milosevic, Z., Cole, J., Gibson, S., Kulkarni, S., Neal, S.: A unified behavioural model and a contract language for extended enterprise. Data & Knowledge Engineering 51(1), 5–29 (2004)
13. Noia, T., Sciascio, E., Donini, F.M., Mongiello, M.: A system for principled matchmaking in electronic marketplace. Intl. Journal of Electronic Commerce 8, 9–37 (summer 2004)
14. Oren, N., Norman, T.J., Preece, A.D.: Argumentation based contract monitoring in uncertain domains. In: Veloso, M.M. (ed.) IJCAI, pp. 1434–1439 (2007)
15. Panagiotidi, S., Vázquez-Salceda, J., Álvarez Napagao, S., Ortega-Martorell, S., Willmott, S., Confalonieri, R., Storms, P.: Intelligent contracting agents language. In: Proceedings of the Symposium on Behaviour Regulation in Multi-Agent Systems -BRMAS 2008, Aberdeen, UK, pp. 49–54 (April 2008)
16. Pitt, J.V., Kamara, L., Sergot, M.J., Artikis, A.: Formalization of a voting protocol for virtual organizations. In: Dignum, F., Dignum, V., Koenig, S., Kraus, S., Singh, M.P., Wooldridge, M. (eds.) AAMAS, pp. 373–380. ACM Press, New York (2005)
17. Sandholm, T., Lesser, V.: Leveled-commitment contracting: a backtracking instrument for multiagent systems. AI Mag. 23(3), 89–100 (2002)
18. Udupi, Y.B., Singh, M.P.: Contract enactment in virtual organizations: A commitment-based approach. In: AAAI. AAAI Press, Menlo Park (2006)
19. Weigand, H., Heuvel, W.: Cross-organizational workflow integration using contracts. Decision Support Systems 33(3), 247–265 (2002)
20. Xu, L.: A multi-party contract model. SIGecom Exch. 5(1), 13–23 (2004)
21. Zuzek, M., Talik, M., Swierczynski, T., Wisniewski, C., Kryza, B., Dutka, L., Kitowski, J.: Formal model for contract negotiation in knowledge-based virtual organizations. In: Bubak, M., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2008, Part III. LNCS, vol. 5103, pp. 409–418. Springer, Heidelberg (2008)

A Video-Based Biometric Authentication for e-Learning Web Applications

Bruno Elias Penteado and Aparecido Nilceu Marana

UNESP - São Paulo State University, School of Sciences, Department of Computing, Bauru
Av. Edmundo Carrijo Coube, 14-01, 17013360, São Paulo, Brazil
{burger,nilceu}@fc.unesp.br

Abstract. In the last years there was an exponential growth in the offering of Web-enabled distance courses and in the number of enrolments in corporate and higher education using this modality. However, the lack of efficient mechanisms that assures user authentication in this sort of environment, in the system login as well as throughout his session, has been pointed out as a serious deficiency. Some studies have been led about possible biometric applications for web authentication. However, password based authentication still prevails. With the popularization of biometric enabled devices and resultant fall of prices for the collection of biometric traits, biometrics is reconsidered as a secure remote authentication form for web applications. In this work, the face recognition accuracy, captured on-line by a webcam in Internet environment, is investigated, simulating the natural interaction of a person in the context of a distance course environment. Partial results show that this technique can be successfully applied to confirm the presence of users throughout the course attendance in an educational distance course. An efficient client/server architecture is also proposed.

Keywords: Biometrics, Web authentication, Face recognition, e-Learning.

1 Introduction

The Internet popularization enabled the development of technologies that leveraged collaborative work. During the past decades it has been seen the proliferation of one of these technologies: web based learning or e-learning systems. Though distance learning has been used long before, by means of correspondence courses, the evolution of information and communication technologies provided an improved and efficient way to deliver contents to remote students. This proliferation is consequence of web based learning characteristics [1]: decreased delivery costs, speed for acquiring knowledge, self-paced learning, geographically open learning, simple updating learning materials, easy management for large groups of students and so on, which allow the students to adapt their schedule and attend their preferred courses regardless from where they are offered.

The e-learning market worldwide, in corporate and the other levels of education, is estimated in over $52 billion by 2010 [2], revealing the extent of this technology.

Nevertheless, some authors [3], [4] point out a serious deficiency: the lack of proper mechanisms for assuring the identity of the remote student. A student might

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 770–779, 2009.
© Springer-Verlag Berlin Heidelberg 2009

give away his credentials to another person to take the course tests, for example. Or the student might provide his credentials to the system and another person takes his place and takes the assessments. So, it is revealed a potential security breach for such systems.

Another important factor to consider is the popularization and, therefore, the decreasing prices of biometric enabled hardware devices, like fingerprint readers, microphones and webcams, many of them embedded in laptops and other sorts of hardware. These facts lead to take biometrics into account as a secure and viable way for remote user authentication in web applications, in particular for web based courses.

2 Biometrics

The Biometrics is an emerging technology that has as basis something you inherently are (anatomically, like fingerprints, face, iris) or do (behaviorally, like typing or signature patterns), as opposed to something you know (password) or something you own (card) [5].

A biometric system is essentially a pattern recognition system that can extract and compare features from fingerprint, face, voice or another characteristic of human body with sufficient uniqueness to differentiate one person from the others. Currently, its main application is in attesting the identification of individuals in commercial and law enforcement applications such as access control to physical environments, homeland security, and electronic commerce among other things.

Some studies were developed in the sense of remote person authentication through the internet [6], [7], but the password-based approach is still predominant for this sort of systems.

There are two modes for biometric authentication: (i) verification, which consists in comparing the biometric trait collected against the credentials provided by the person, checking if he is who he claims to be and (ii) identification, which attempts to check if the user is enrolled and whose sample is matched from the database, comparing the biometric trait without further credentials.

2.1 Face Recognition

Among the most studied biometric technologies is the face recognition modality. It may be stated as: given still or video images of a scene, identify one or more persons in the scene using a database of pre-enrolled individuals [8].

Face recognition performance is not as accurate as fingerprint or iris recognition, but its acceptability and easiness of collection are distinctive.

The face recognition process consists of the following modules: the first step involves the segmentation of the face from a cluttered background, subtracting the image portion which contains the face. In the second step some measures are taken (distances, coefficients, etc.) and these measures are used to represent the face so that it can be compared more computationally efficiently to another. The final step, the recognition, uses the measures taken in the last step and performs some matching scheme to identify or verify the individual's identity.

2.1.1 Face Recognition from Still Images

The performance of face recognition systems based on still images is affected more deeply by the following issues: pose, expression and illumination variations and by the use of accessories that may occlude part of the face, what demands robust transformations and techniques in order to alleviate these problems.

Some algorithms developed to handle still images of faces, like PCA [12] and LDA [13], present good results, especially in collaborative scenarios, but all of them suffer with the problems stated previously.

2.1.2 Face Recognition from Video

Other methods for face recognition are based on video analysis. Since a movie clip is fundamentally composed by a series of still images (frames), displayed several times per second, traditional face recognition algorithms may be applied to this collection of images, frame by frame.

By exploring this feature of videos, some additional properties can be extracted [10]: (i) observation set: frames are considered as a set of images (observations) of a same individual; (ii) dynamic/temporal continuity: use of models which take into account the posterior probability to describe the face and head variations as a whole along the sequence; (iii) 3D models: reconstruction of a model for the individual from a group of multiple frames.

There are three approaches for face recognition using images and video combination [11]: (i) image-to-image: both query and database samples are still images; (ii) video-to-video: both query and database are video samples and (iii) image-to-video: query sample is a video compared against a database of still images.

Although the face recognition from video presents some advantages over the still approach, it also poses some difficult challenges. Most of videos collected are of poor quality and low resolution, without the person's collaboration (i.e. non-frontal poses), with big variations of illumination and facial expressions, with uncertainty in detection due to motion, and so on.

Systems based on recognition from video are of great interest, given the video cameras installed base and embedded in new products. Surveillance, activity analysis and user interaction are some examples of applications for such systems.

3 Proposed System

To study the viability of web authentication using biometric recognition, it was developed a prototype system that processes video input from the user interacting with an e-learning environment, through a webcam, and use it to attest his credentials (verification mode), expecting to improve the efficiency of remote user authentication.

3.1 Face Detection

Face detection consists in finding texture patterns in the image that are likely to be a face. The face detection module of the system is based on the Viola-Jones algorithm [14]. It works by trying to find features in the image that code some information of the class to be detected. To accomplish this task, Haar-like features are used: they are

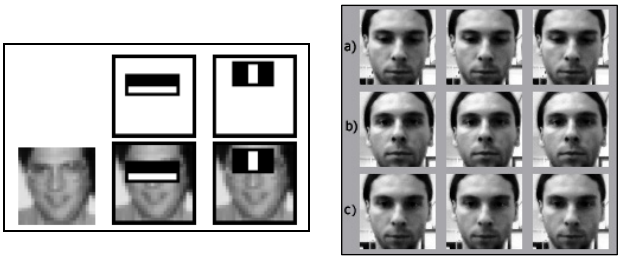


Fig. 1. *Left*: relationships between Haar-like features and face contrasts (Viola, Jones, 2004). *Right*: selected images of an individual in the following poses: a) reading, b) frontal, c) typing.

responsible for coding information about the presence of contrasts between image regions. For face detection, the natural face contrasts and their spatial relationships are explored, as in Figure 1a.

The main feature of this algorithm is the speed in which the faces are processed (detected), even in different scales. It uses sub windows of different sizes that slide through the image. The decision if the sub window belongs to the class is composed by several simple phases. In case the sub window is rejected in one of these phases, it is rejected at all, stopping the process for that sub window and moving to another.

3.2 Feature Extraction and Face Recognition

The feature extraction module is based on the Principal Component Analysis (PCA) technique. Developed by the studies of Turk [12] and Kirby [15], this technique consists of finding the eigenvectors that constitute the face sub-space bases (eigenspace) obtained by the covariance matrix formed by the correlation between pixels of a training set. In summary, each image of a human face in the database is represented in terms of a linear combination of these eigenvectors, and these coefficients will be the new face representation in the eigenspace.

Once the face is detected in the previous step, it is decomposed in the coefficients that represent the face, based on the eigenvectors obtained in the training phase, representing a point in this n-dimensional eigenspace. These coefficients are then used as feature vector to represent the face of the individual.

The features are passed on to the next module, which iterates through the database of enrolled individuals and calculates the distance between the feature vector sampled and the template of the individual, stored during the enrolment phase. The recognition is performed using distance-based classifier. In this work, Euclidean distance was used.

3.3 System Architecture

Due to the computational complexity of the algorithms, some points must be taken into account when deploying the system for a large number of users. The system load must be distributed between the client and the server to reach a reasonable response time. In this work, the following items were attempted:

(i) reduce network traffic, by transmitting only the feature vectors instead of the video streaming;
(ii) reduce the server load used to process resource intensive algorithms, by giving the client the processing task of tracking and extracting the features;
(iii) security permissions, by allowing the webcam to be captured in the client user.
(iv) integrate to any LMS (Learning Management System), independently of the technology it was built upon, by using a wrapper in accessing the web system;

Figure 2 shows how the modules communicate.

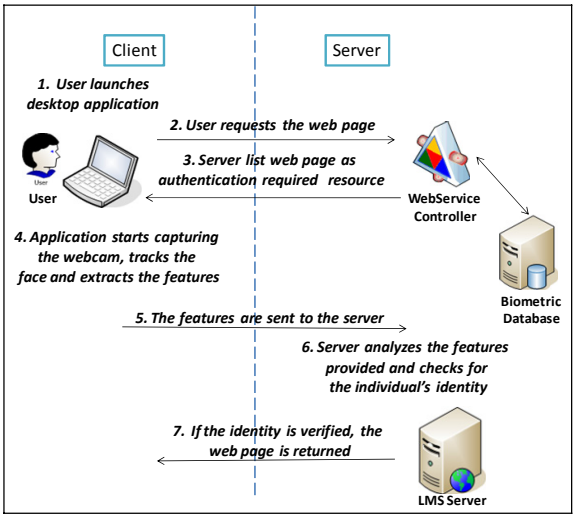


Fig. 2. Client and server modules and workflow of the proposed system

In this schema, there are the following subsystems: (i) the desktop application, which blocks or allows the access to the course and captures and extracts the feature vector to be sent to the server; (ii) the LMS Server, which hosts the web based course; (iii) a web service controller, which manages if a web page being requested is marked to be verified and processes the feature vectors collected; and (iv) the biometric database, used to store the user's templates. This way, the previous items can be satisfied.

The user, in order to attend the course, launches a desktop application in his local computer. Then, the application requests for the web application in the server. Along with the request, a query is also sent to verify if the web page being requested requires the biometric authentication. This might be the case for an assessment or a protected content, for instance. If it is not, it is rendered back to the client. Otherwise, the client application starts capturing the video using the user webcam. While the application processes the video, detecting, extracting and pre-processing the face portion of the frame, it sends the feature vector asynchronously to the web service controller. The web service queries the database for the biometric trait and returns the answer to the desktop client. The desktop application then blocks or renders the web page.

4 Experiments

In order to evaluate the proposed method for biometric authentication in a distance course web application, the following methodology was adopted.

In the first video collecting session, it was asked for the individuals to browse in a certain website and answer one written question in a webpage. The individuals were oriented to vary their facial poses and expressions during the session. Forty five samples were collected, one per person, and with duration of one minute and a half, in average. The sampled frames of every video collected in this session were used by the PCA training algorithm in order to build the set of eigenvectors.

The second session was performed some weeks after the first. In this session, the same individuals were oriented to do the following steps to simulate the browsing behavior in an e-learning system: (i) look frontally to the webcam for some seconds; (ii) read a 500 characters length text; (iii) fill in a form with some personal data. Five seconds fragments were manually cropped out of the videos, one for each pose.

The training and test sets were done using a Creative Webcam Pro eX PD1050 webcam. Both sets were collected without large illumination variations in the environment.

For the database construction, three frames were chosen in order to represent each pose and individual (nine in total). For this choice, it was used the method proposed by [16], which selects the most varying images inside a set of images. Figure 1b shows the selected images for one of the individuals in the experiment.

The images were sampled frame by frame and the faces were detected and extracted using Viola-Jones algorithm implemented in the OpenCV library [17]. As preprocessing step, the extracted images were resized to a standard dimension (64x64 pixels) and histogram equalization was applied to adjust its contrast.

In order to have a single decision for a given set of frames analyzed, it is needed to fuse the results given by every frame. To fuse the results obtained from individual frames, a majority-voting scheme was adopted. Thus, it is assigned to the face the identity which most of the decisions agree, handling every decision as an independent event. As the decisions are given by the same algorithm, no other weighting or normalization is needed.

5 Experimental Results

To measure the system performance, the following analyses were carried out:

a) Face detector algorithm efficiency. Faces must be extracted from the frames where they are in. Hence, the global performance of the system depends on this step (detection).

b) Number of eigenvectors selected and the respective recognition rates. This is important because it impacts on the computational effort spent during the authentication. Being a web application, under the client/server architecture, it is required to have a low response time.

c) Recognition rate related to top N users returned from the database. This information is useful in biometric authentication, where the user claims who he is, and the system searches if the claimed identity is in the top N matches.

Regarding Viola-Jones algorithm evaluation, the following items were considered: minimum sub window dimensions from which the face will be searched from, number of false positives and negatives and the processing time for the samples. These measurements were taken off-line.

Figure 3 shows the time spent to locate faces in function of the sub window dimensions. These dimensions serve as starting point from which the face detection can be made. Face patterns smaller than these sub window dimensions are ignored. Faces of users sitting by the computer used to have more than 100x100 pixels, in total of 320x240 of resolution for the frames collected. With a 100x100 sub window, the processing is real time.

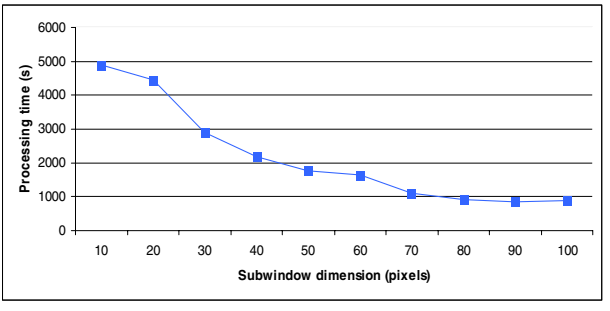


Fig. 3. Processing time over the sample set in function of window scanning size

This also carries other consequences: decreasing number of false positives (accepted non-faces) as well as an increasing number of false negatives (face rejections). Figures 4 and 5, respectively, show these facts.

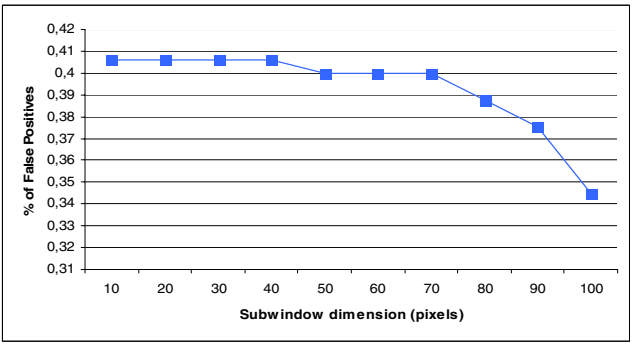


Fig. 4. Non-face patterns mistakenly classified as faces, considering the sample set

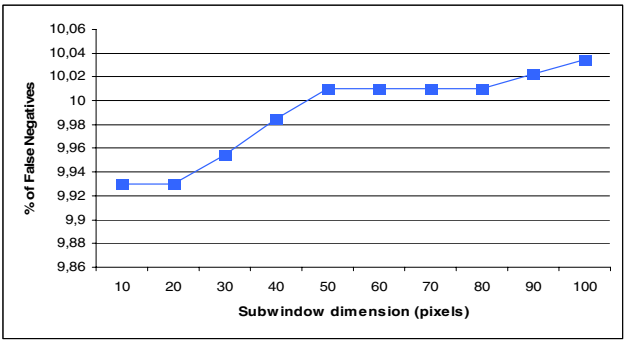


Fig. 5. Undetected faces, considering the sample set

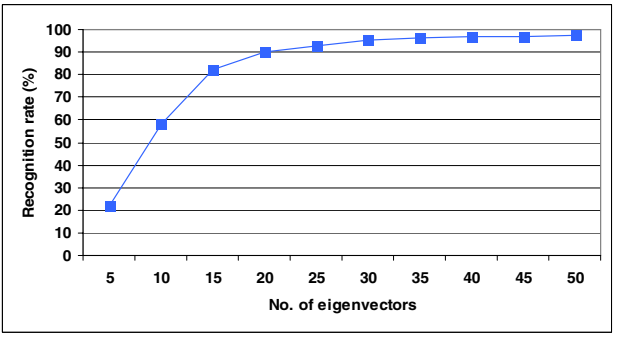


Fig. 6. Recognition rate considering the number of eigenvectors used to represent the face

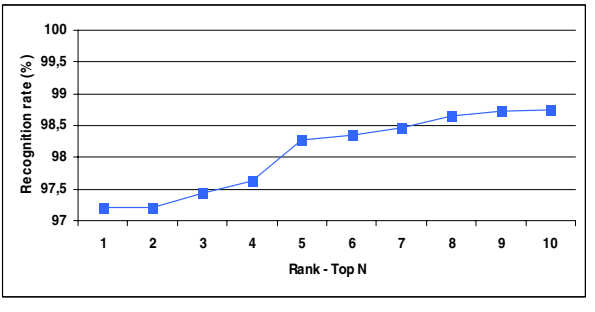


Fig. 7. Recognition rate considering the top N matches in frame level (using 50 eigenvectors)

As the false positive rate influences negatively the system performance, it is desired to keep it as low as possible. Regarding the false negative rate, it is expected not to influence the system overall performance, because of the information abundance. Thus, the system can work with larger sub windows, providing low response times while keeping the recognition rates.

Another factor influencing the system processing time is the number of eigenvectors used to represent the face template, the feature vector length. As depicted in

Figure 7, from 25 eigenvectors on, the recognition rate tends to increase just a little; still, keeping a good level of recognition rate.

Another important measurement is the number of identities needed so that the system can correctly authenticate the individual. Figure 7 shows the top N matches for the selected algorithm. It can be noted that even for the first identity returned (top 1) it presents a high level rate, considering the total amount of frames used.

6 Conclusions

In this work we explored an alternative technological approach for the problem of remote authentication in the context of an e-learning web environment. Biometric authentication can help assuring that the actual individual is taking the course. To accomplish this task, the natural interaction of a user was taken into account, attempting to reproduce his behaviour during the course attendance. Because of the client/server architecture nature, it was also designed an efficient architecture, in order to reduce the computational load and traffic between the client and server stations, independently of the LMS employed. The partial results show that the face tracking module performs well both in processing time and correct matches and the recognition module shows reasonable results. The techniques employed, though not novels, can be still efficiently applied to this problem. As future works are the use of an algorithm to better explore the temporal information from video and the how to better fuse the individual frame decisions.

References

1. Cantoni, V., Cellario, M., Porta, M.: Perspectives and Challenges in e-Learning: Towards natural Interaction Paradigms. Journal of Visual Languages and Computing (15), 333–345 (2003)
2. Global Industry Analysts, eLearning: A Global Strategic Business Report (2008)
3. Marais, E., Argles, D., von Solms, B.: Security issues specific to e-assessments. In: 8th Annual Conference on WWW Applications (2006)
4. Rabuzin, K., Baca, M., Sjako, M.: E-learning: biometrics as a security factor. In: Proceedings of the International Multi-Conference on Computing in the Global Information Technology, pp. 64–74 (2006)
5. Miller, B.: Vital signs of identity. IEEE Spectrum 31, 22–30 (2004)
6. Jain, A.K., Prabhakar, S., Ross, A.: Biometrics-based web access. Transactions of the Institute of British Geographers 7, 458–473 (1998)
7. Kounoudes, A., Kekatos, V., Mavromoustakos, S.: Voice Biometric Authentication for Enhancing Internet Service Security. Information and Communication Technologies ICTTA 1, 1020–1025 (2006)
8. Chellappa, R., Wilson, C.L., Sirohey, S.: Human and machine recognition: a survey. Proceedings of the IEEE 83(5), 705–741 (1995)
9. Zhao, W., Chellappa, R., Phillips, P.J., Rosenfeld, A.: Face Recognition: A Literature Survey. ACM Computing Surveys, 399–458 (2003)
10. Zhou, S.K., Chellappa, R., Zhao, W.: Unconstrained Face Recognition. Springer, New York (2006)

11. Phillips, P.J., Moon, H., Rivzi, S., Rauss, P.: The FERET Evaluation methodology for face recognition algorithms. IEEE Transactions on Pattern Analysis and Machine Intelligence 22, 1090–1104 (2000)
12. Turk, M., Pentland, A.: Eigenfaces for recognition. Journal of Cognitive Neuroscience 3(1), 71–86 (1991)
13. Belhumeur, P.N., Hespanha, J.P., Kriegman, D.J.: Eigenfaces vs. Fisherfaces: recognition using class specific linear projection. IEEE Transactions on Pattern Analysis and Machine Intelligence 19(7), 711–720 (1997)
14. Viola, P.A., Jones, M.J.: Robust real-time object detection. International Journal of Computer Vision 57(2), 137–154 (2004)
15. Kirby, M., Sirovich, L.: Application of the Karhunen-Loéve procedure for the characterization of human faces. IEEE Transactions on Pattern Analysis and Machine Intelligence 12(1), 103–108 (1990)
16. Thomas, D., Bowyer, K.W., Flynn, P.J.: Multi-frame approaches to improve face recognition. In: IEEE Workshop on Motion and Video Computing (2007)
17. Open CV: Open Source Computer Vision Library (2008), http://opencvlibrary.source-forge.net/

Modeling JADE Agents from GAIA Methodology under the Perspective of Semantic Web

Ig Ibert Bittencourt[1,2], Pedro Bispo[2], Evandro Costa[2], João Pedro[2], Douglas Véras[2], Diego Dermeval[2], and Henrique Pacca[2]

[1] Computer Science Department, Federal University of Campina Grande, Paraíba, Brazil
ibert@dsc.ufcg.edu.br
[2] Computing Institute, Federal University of Alagoas, Maceió, Brazil
GrOW - Group of Optimization of the Web
pedrobssantos@gmail.com, evandro@fapeal.br, joaopedro.plima@gmail.com
http://www.grow.ic.ufal.br

Abstract. Building multi-agent software systems is pointed out as a high complex task and researchers have raised different issues for building several applications. Therefore several AOSE methodologies and MAS frameworks have been proposed to facilitate the hard task of modeling and building high complex systems. However, those methodologies in an attempt to model complex systems end up being hard to use and to ensure the consistency between each part. On the other hand, ontologies have been considered useful for representing the knowledge of software engineering techniques and methodologies in order to provide an unambiguous terminology that can be shared, reusable, and ensure the consistence between the concepts involved. This paper proposes ontologies for specifying agents through the use of GAIA methodology, JADE Framework and SWRL rules to map the instances from GAIA ontology to JADE ontology. Finally, it is presented a case study and a discussion to demonstrate their use.

Keywords: Agent-oriented software engineering, Agent methodologies, JADE framework, Ontologies, A semantic web rule language, GAIA methodology.

1 Introduction

Building high complexity multi-agent software systems, especially the domain-oriented ones (i.e. e-commerce and e-learning systems), is pointed out as a very complex task because several aspects must be considered and put together, such as roles, interaction protocols and services. In addition, there is a considerable gap between the analysis, design and implementation phases. Moreover, researchers have raised several issues for building several applications, such as high development costs, complexity to develop artificial intelligence techniques, scalability, content sharing and so on.

Based on the mentioned reasons, software engineers and researchers [1] [2] [3] have decided to decompose the system as a promise to ease the development and maintenance processes. Although the system decomposition improves the development and maintenance of complex systems, it is not enough to ensure a system's high-quality.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 780–789, 2009.
© Springer-Verlag Berlin Heidelberg 2009

As a result, [4] pointed out that analyzing, designing, and implementing complex software systems as a collection of interactive and autonomous agents assure, to software engineers, a number of significant advantages over contemporary methods.

Due that several agent-oriented software engineering (AOSE) methodologies (such as GAIA [5], Aalaadin [6], Moise+ [7], TROPOS [8] and others) and multi-agent-based (MAS) frameworks (such as JADE [9], JADEX [10], and others) were created in order to facilitate the hard task of modeling and building high complex systems. However, those methodologies, in an attempt to model complex systems, end up being hard to use and to ensure the consistency between each part (roles, protocols, services, resources, agents). As a matter of fact it is not guaranteed to achieve a concordance between the modeling and project phases of such systems.

As [11] stated, the process of designing a system consists in instantiating the system meta-model that the designers have in their mind in order to fulfill the specific requirements. Despite the fact that a meta-model is used, it is also necessary to define the language representation of meta-models to specify as the AOSE methodology as the MAS framework. Indeed, ontologies provide an unambiguous terminology that can be shared by all involved in a software development process. They can also be as generic as needed allowing its reuse and easy extension. These features turn ontologies useful for representing the knowledge of software engineering techniques and methodologies [12].

As a result, these ontologies combined appropriately (through the use of rules defined in SWRL to map them) make possible the improvement of the development of those systems, by diminishing the difficulties mentioned above.

This paper proposes i) ontologies for specifying agents through the use of GAIA Methodology and JADE Framework and ii) SWRL rules to map the instances from GAIA ontology to JADE ontology. Finally, it is presented a case study and a discussion to demonstrate the use of the ontologies and rules.

2 Ontologies

This section describes the GAIA [1] and the JADE ontology [2] partially, along with the SWRL rules and a short description.

Environment. Multi-agents systems are settled on an environment, and so the identification of the environment is the starting point for MAS modeling. Therefore, the system's available resources, sub-organizations and relational rules are to be made as explicit as possible. The environment has a set of resources that is used by agents in the systems, and permissions like read, write or consume are associated with these resources. For every permission, an interval is defined in order to designate the upper and lower bounds. In other words, values that represent the permission coverage of the resource.

Sub-organizations. Sub-organizations divide the system in organisms with sub-goals, aim at modularizing, diminish its complexity and to make its management easier. Basic responsibilities should be identified in every sub-organization, exposing agents that

[1] The GAIA ontology is available at http://grow.ic.ufal.br/owl/Agents/GAIA/GAIA.owl

[2] The JADE ontology is available at http://grow.ic.ufal.br/owl/Agents/JADE/JADE.owl

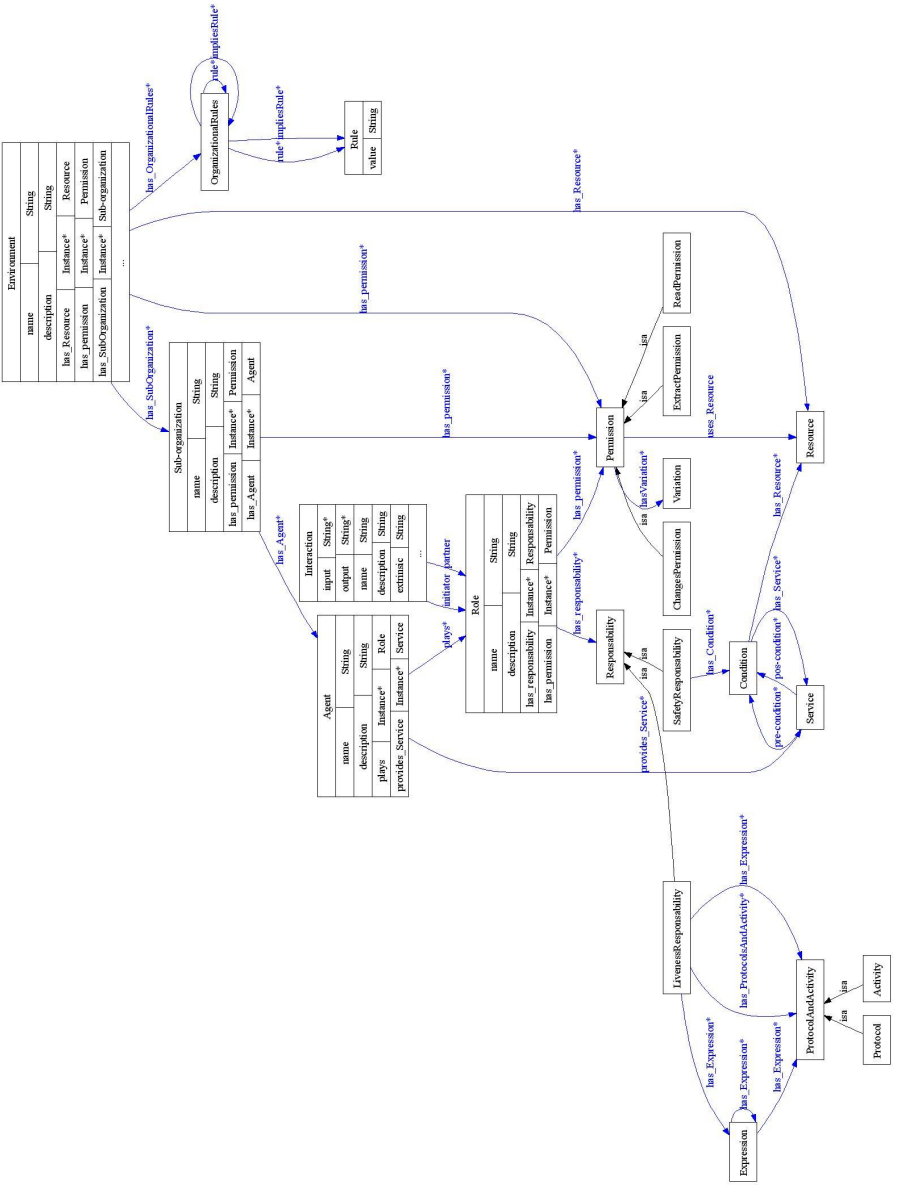


Fig. 1. GAIA Ontology

will be part of it. The agent is defined as a living entity that contains a list of roles and set of services that he can provide. Services have inputs and outputs, pre and post-conditions. In addition, every role has a set of permissions for resources in the environment, and with a series of responsibilities and interactions. When a role has dependency relationship, needing others to concretize its goals, interactions are used so that roles

can communicate in order to achieve a common goal. For a role to accomplish its objectives, he must obey a set of responsibilities, distributed in two categories: Liveness and Safety. On the one hand, Liveness responsibility defines a set of expressions that must be followed so that a responsibility can be accomplished. On the other hand, Safety responsibilities create restrictions over the environment's resources in a way that one action can only be performed if it complies with the conditions imposed by the safety responsibilities.

Organizational Rules. Organizational rules model the restriction that agents must follow while living, a very import characteristic in open systems. A system must contain an internal regiment, in addition of being able to set its rules for every new entity that may eventually be inserted into the environment. These rules, then well defined, can stop malicious entities to compromise the stability, instituting a conduct policy that must be strictly followed by those who inhabit the environment.

Table 1. GAIA-JADE rules

Origin concept	Destination concept	SWRL Rule
gaia:Agent	JADE:JADEAgent	swrl-AgentToJADEAgent
gaia:Service	JADE:Service	swrl-ServiceToService
gaia:Interaction	JADE:ACLMessage	swrl-InteractionToACLMessage
gaia:Activity	JADE:Behaviour	swrl-ActivityToBehaviour

Table 2. GAIA-GAIA rules

Origin concept	Destination concept	SWRL Rule
gaia:LivesnessReponsability	gaia:Service	swrl-LivenessResponsabilityToService
gaia:LivesnessReponsability	gaia:Service	swrl-LivenessResponsabilityToOutput

2.1 JADE Ontology

The JADE ontology (see Figure 2) describes agents according to the structure proposed by the framework, used for developing agents in Java. It makes the agent-based development easier by using an environment that implements characteristics required by the agent methodology, such as lifecycle and intercommunication. Additionally, it has a very rich graphical suit. A short description of the classes present on the ontology is shown in the following subsections.

AgentPlatform Class. The platform represents the application core, on that all agent's sub-organizations and all agents living on the enviroment are concentrated. This platform is made by a set of containers that sub-divide the system into organizations, that in its turn contain agents with similar goals.

JADEAgent Class. Agents are considered as living entities that have reacting and proactive behaviours. These agents can interact with other entities through ACLMessages, which are mecanisms of negotiation or even coordination between agents. Agents provides services that can be used to achieve its goals or to negotiate with other entities.

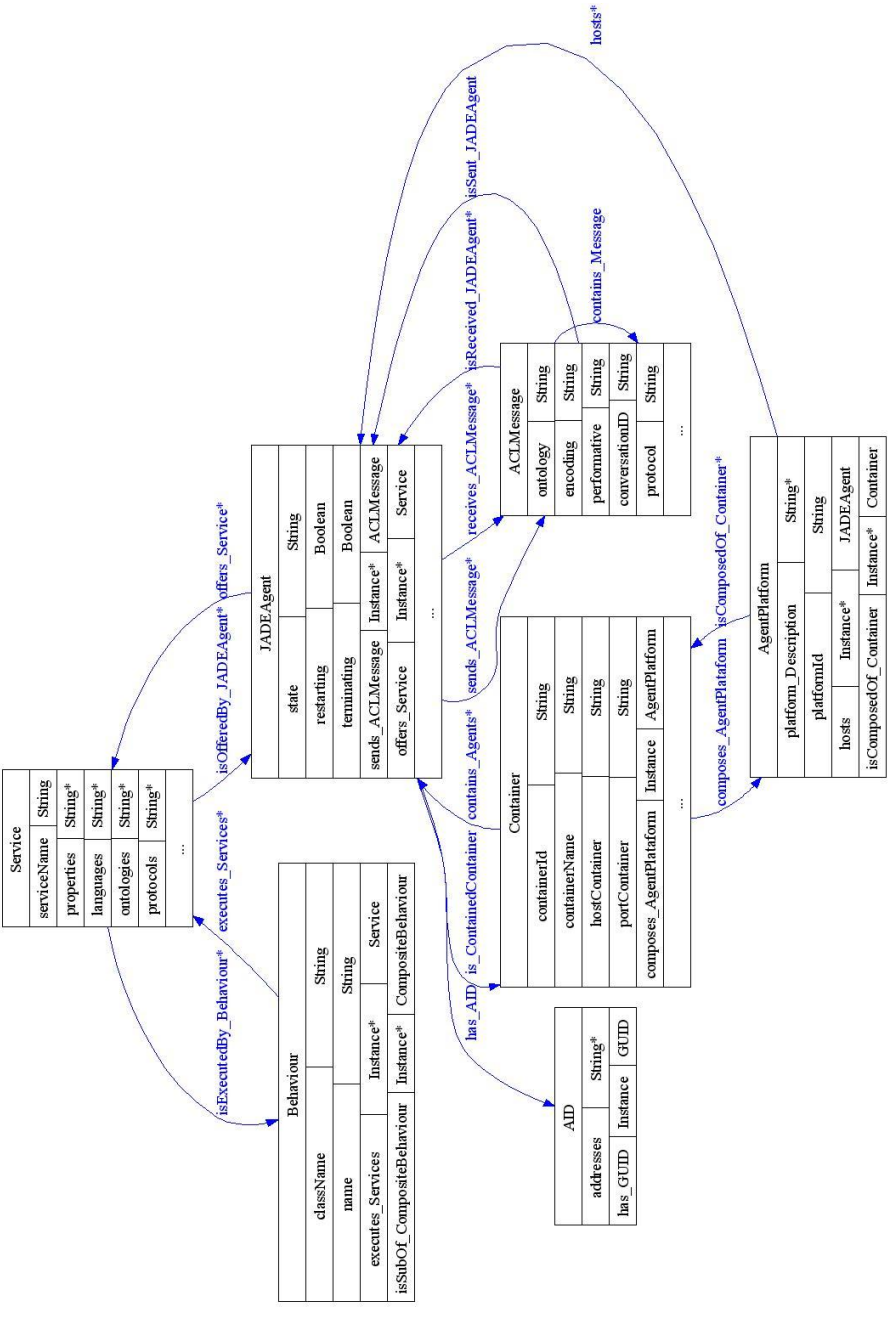


Fig. 2. JADE Ontology

ACLMessage Class. ACLMessages are means for interaction among agents and they are standardized by FIPA. These messages specifies all protocols of interaction between agents and how these agents should react when a order or a information is received.

Behaviour Class. Behaviours represents tasks that can be accomplished by the agents. In order to achieve its goals, an agent can execute one or more behaviours. These can be seen as services that represents a level of abstraction for service discovery. Services are registered in yellow pages, and work similar to a big service provider.

2.2 GAIA-JADE Mapping

SWRL is based on a combination of the OWL DL and OWL Lite sublanguages of the OWL with the Unary/Binary Datalog RuleML sublanguages of the Rule Markup Language. The proposed rules are of the form of an implication between an antecedent (body) and consequent (head). Both the antecedent and consequent consist of zero or more atoms. Based on the former ontologies, SWRL rules were created to map instances of one ontology into the other. The table 1 shows the mapping of concepts and the rules that are described below:

swrl-AgentToJADEAgent. In this rule, a GAIA agent is mapped into a JADE agent. To do this, GUID is created and its local name property is set as the GAIA agent name. The GUID is bound to an AID that in its turn is set as the JADE agent property.

swrl-ServiceToService. With this rule, a GAIA service is mapped into a JADE service. This rule focuses on all JADE agents that were mapped from GAIA agents. For each service provided by the GAIA agent, a JADE service is created and bound to the corresponding JADE agent, respecting all its properties.

swrl-InteractionToACLMessage. This one relates GAIA interactions with ACL messages from the JADE framework. Basically, the interactionś initiator is mapped into the interaction sender, and the interactionś partner is mapped into the message receiver.

swrl-ActivityToBehaviour. A JADE behavior is instantiated for every GAIA activity and a JADE service with the same name of the GAIA Liveness Responsability. Finally it is set as the JADE service is executed by the instantiated behavior.

swrl-LivenessResponsabilityToService. This rule creates a link between the liveness-Responsability and the service, both of them from GAIA. The service name is the same as the Liveness Responsability. Liveness responsability's protocols that are set to true are mapped into service's input.

swrl-LivenessResponsabilityToOutput. works similar to the swrl-LivessResponsabilityToService, except that Liveness responsability's protocols which are set to false are mapped into outputs instead.

3 Case Study

The goal of this section is to validate and describe the applicability of the present proposal, by modeling a multi-agent educational environment called ForBILE [13]. All the

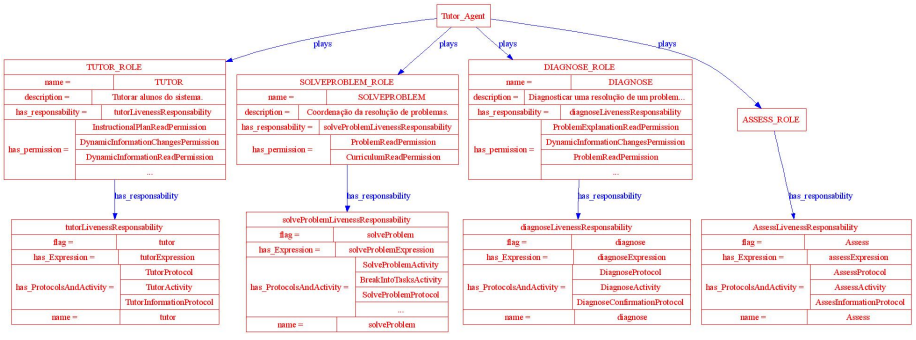


Fig. 3. Tutor_Agent Instance

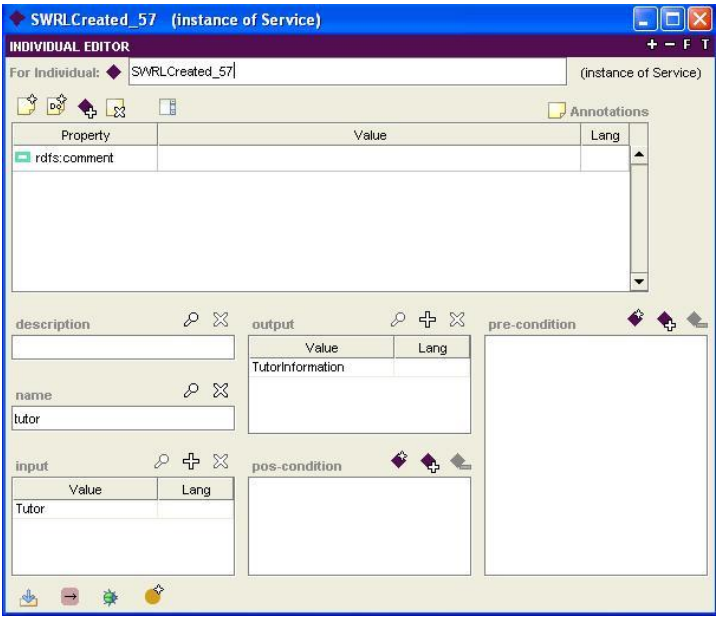


Fig. 4. Tutors Services mapped from livenessResponsability

instantiation is be done using Protégé OWL Editor [14]. The three following steps exemplify the needed phases for the construction of multi-agent systems according to the former proposed ontologies.

Instantiation of the GAIA Ontology. The figure 3 shows the instantiation of an Agent that exists on the ForBILE, and two of its roles. The figure 3 shows TutorAgent, an agent that plays the ASSESS, DIAGNOSE, SOLVEPROBLEM and TUTOR roles, besides, it also shown in the figure the responsibilities of every role.

Execution of GAIA Rules. By executing the rules, the semi-automatic mapping of the LivenessResponsabilities into services is completed. The figure 4 shows

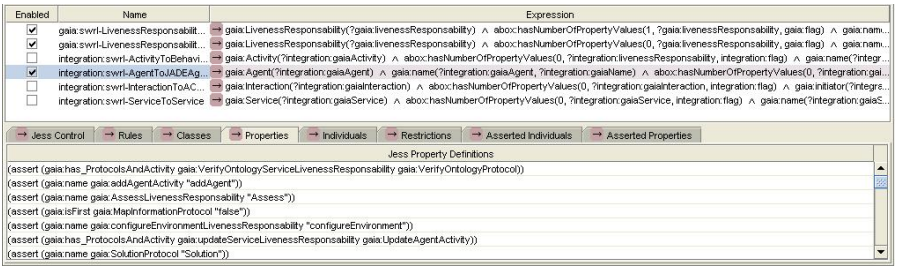


Fig. 5. SWRL Rules

TutorAgent and the SolveProblem service that was mapped for the agent. The responsibilities mapping into services is semi-automatic, because it is still needed to fill the pre and pos conditions, if they exist.

Execution GAIA-JADE Integration Rules. In this step, the SWRL rules are used to map the GAIA MAS modeling into JADE behaviours.

This stage occurs also in a semi-automatic way, because there are still additional configuration that needs to be filled, such as Platform e Address, due to these attributes not belonging to the GAIA methodology scope(see Figure 5).

Even though a agent and its services were modelled in this section, it is important to note that the entire ForBILE modeling were covered with respect to the GAIA models and they can be found at http://www.grow.ic.ufal.br/copy_of_projetos/forbile-1/documentacao/gaia/requisitos.

4 Related Works

This section addresses to papers related to the present one. The goal here is looking for other papers that use ontologies for semantically represent the meta-models of existing methodologies and find other works that automatically map agents from the analysis phase to the project.

According to [15] a system is built based on the GAIA methodology and using the JADE framework to make the implementation. This approach does not support the use of ontologies to automatically generate JADE agents specifications.

[12] suggests ontologies to model multi-agent system, giving support to agent organization on the environment. Despite of that, this approach does not cover with the system's project phase nor with the automatic mapping of the entities modeled by the GAIA ontology.

In [16], the authors try to find a way of uniting three different multi-agent systems modeling methodologies (ADELFE, GAIA and PASSI) by studying its meta-models and the concepts related to them, and making a unified meta-model for building MAS. However, the authos do not express the GAIA ontology in a complete way and also do not map it to an implementation framework. In [17], the authors tryed to establish a strategy for identifying a common meta-model that could be vastly adopted by all

the Agent Oriented Software Engineering society, however, this proposal has the same problem of prior one.

Even though the cited researches are very important to the modeling phase of multi-agent systems, they do not worry about the system's design phase and consequently did not provide aim at providing a way to automatically map these phases and decrease the gap between them.

4.1 Discussion

This section aims at discussing some aspects with regards to the development of the ontologies, rules, and case study. The goal was to make the specification (analysis and design phases) of multiagent systems easier, through the use of ontologies and rules. The ontologies provided an unambiguous terminology that can be shared, reusable, and ensure the consistence between the concepts involved. In addition, the rules hided several implementation details because they provide the mapping between the specification defined in GAIA ontology to the JADE ontology. Some important aspects are discusses as follows.

- Time to specify the Multiagent System: the cost to specify the system is decreased because the knowledge engineers can use the ontologies and rules and verify the consistence of the specification;
- Completeness: it was possible to specify both methodology and JADE framework through the use of OWL ontologies. However, several parts were complicated to specify in some parts of GAIA. For instance, in GAIA the specification of rules, conditions, and expressions. On the other hand, the specification of JADE ontology had no expressiveness problem;
- Mapping: the rules were used to ensure the mapping between the ontologies. SWRL was used as a natural solution to integrate with OWL ontology. However, some problems to express the mapping were complicated because SWRL has several limitations (such as disjunction and negation are excluded, no block structure is supported, the SWRL rule head "makes" its atoms true, and so on). In addition, it was not possible to map 100% of the specification because there are some parts of JADE that are very specific, such as the platform which the agents are performed.

5 Conclusions

This paper presented an Ontology-based approach for modelling JADE agents from GAIA methodology. Were created an ontology for specifying the GAIA methodology and JADE Framework. In addition, rules to ensure the relation between both ontologies were defined. OWL, SWRL, Protégé , and JESS were the technologies used. The contributions of the proposed paper are the ontologies and SWRL rules to define the methodology and framework and to ensure the relation between them. This approach makes easier and fast the analysis and design of multi-agent systems by using GAIA and JADE. As future works, it is necessary to build new rules to map others concepts presents in GAIA, such as pre and pos conditions.

6 Additional Authors

Tamer Cavalcante (tamersgc@gmail.com), Lucas Braz (lukas.braz@gmail.com), Rafael Ferreira (rafaelflmello@gmail.com), Heitor Barros (rotieh@gmail.com), Marlos Silva (marlos.tacio@gmail.com), Tarsis Toledo (tarsiswt@gmail.com), Alan Silva (alanpedros@gmail.com), Ibsen Mateus Bittencourt (ibsen.bittencourt@gmail.com), Willy Tiengo (willy.tiengo@gmail.com).

References

1. Wooldridge, M.: Multi-Agent System: An Introduction to Distributed Artificial Intelligence. John Wiley Sons Ltd., Chichester (2002)
2. Weiss, G.: Multiagente systems - A modern approach to distributed modern approach to artificial intelligence. The MIT Press, Cambridge (1999)
3. Scott, A., Deloach, M.F.W., Sparkman, C.H.: Multiagent systems engineering. International Journal of Software Engineering and Knowledge Engineering (IJSEKE) 11, 231–258 (2001)
4. Jennings, N.R.: An agent-based approach for building complex software systems. Communications of the ACM 44, 35–41 (2001)
5. Zambonelli, F., Jennings, N.R., Wooldridge, M.: Developing multiagent systems: The gaia methodology. ACM Trans. Softw. Eng. Methodol. 12, 317–370 (2003)
6. Ferber, J., Gutknecht, O.: Aalaadin: a meta-model for the analysis and design of organizations in multi-agent systems. In: ICMAS 1998 (1998)
7. Hubner, J.F., Sichman, J.S., Boissier, O.: A model for the structural, functional, and deontic specification of organizations in multiagent systems. In: Bittencourt, G., Ramalho, G.L. (eds.) SBIA 2002. LNCS, vol. 2507, pp. 118–128. Springer, Heidelberg (2002)
8. Giunchiglia, F., Mylopoulos, J., Perini, A.: The tropos software development methodology: Processes (2001)
9. Bellifemine, F.L., Caire, G., Greenwood, D.: Developing Multi-Agent Systems with JADE. Wiley Series in Agent Technology. John Wiley & Sons, Chichester (2007)
10. Lars Braubach, A.P., Lamersdorf, W.: Jadex: A BDI-Agent System Combining Middleware and Reasoning. Birkhäuser, Basel (2005)
11. Wooldridge, M., Ciancarini, P.: Agent-oriented software engineering: The state of the art, pp. 1–28. Springer, Heidelberg (2001)
12. Girardi, R., Leite, A.: Ontomadem: An ontology-driven tool for multi-agent domain engineering. SEKE, Knowledge Systems Institute Graduate School, 559–564 (2007)
13. Bittencourt, I.I., de Barros Costa, E., Silva, M., Soares, E.: A computational model for developing semantic web-based educational systems. Knowledge Based System, Special Issue on Artificial Intelligence and Blended Learning (2009)
14. Stanford (2008), http://protege.stanford.edu/overview/protege-owl.html
15. Moraitis, P., Spanoudakis, N.: Combining gaia and jade for multi-agent systems development. In: Proceedings of the 17th European Meeting on Cybernetics and Systems Research (2004)
16. Bernon, C., Cossentino, M., pierre Gleizes, M., Turci, P., Zambonelli, F.: A study of some multi-agent meta-models, pp. 62–77. Springer, Heidelberg (2004)
17. Bernon, C., Cossentino, M., Pavón, J.: Agent-oriented software engineering. Knowl. Eng. Rev. 20, 99–116 (2005)

A Business Service Selection Model for Automated Web Service Discovery Requirements

Tosca Lahiri and Mark Woodman

Middlesex University e-Centre, The School of Engineering and Information Sciences
The Burroughs, Hendon, London NW4 4BT, U.K.
tl173@live.mdx.ac.uk, m.woodman@mdx.ac.uk

Abstract. Automated web service (WS) discovery, i.e. discovery without human intervention, is a goal of service-oriented computing. So far it is an elusive goal. The weaknesses of UDDI and other partial solutions have been extensively discussed, but little has been articulated concerning the totality of requirements for automated web service discovery. Our work has led to the conclusion that solving automated web service discovery will not be found through solely technical thinking. We argue that the business motivation for web services must be given prominence and so have looked to processes in business for the identification, assessment and selection of business services in order to assess comprehensively the requirements for web service discovery and selection. The paper uses a generic business service selection model as a guide to analyze a comprehensive set of requirements for facilities to support automated web service discovery. The paper presents an overview of recent work on aspects of WS discovery, proposes a business service selection model, considers a range of technical issues against the business model, articulates a full set of requirements, and concludes with comments on a system to support them.

Keywords: Web Services, Discovery, Interoperability, Requirements, Business Service Model.

1 Introduction

The automated discovery and use of a web service (WS) i.e. programmed location, selection and use without human intervention, is a goal of service-oriented computing [1]. However, as discussed below, only partial solutions have so far been attempted.

Considerable work has been carried out to address the many aspects of automated web service discovery. However, no comprehensive, coherent set of requirements appears to have been published; only small subsets have been considered as solutions to automated WS discovery. Clearly, a full set of requirements is needed if a viable solution is to emerge, one that takes account of interacting and conflicting requirements. We aim to articulate a full set of requirements from which it would be possible to build an appropriate mechanism (e.g. an infrastructure layer) that meets the requirements for automated WS discovery. It is particularly important to look at the whole requirements picture rather than just focusing on a particular problem facing web service discovery, because many requirements may be in conflict with each other.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 790–803, 2009.
© Springer-Verlag Berlin Heidelberg 2009

This paper is organized as follows: we first review the scenarios for using web services and establish an appropriate perspective for automated discovery; next we examine the current technical solutions in WS discovery research, noting the requirements they implicitly or explicitly address; we then investigate a model for business service procurement as a guide for WS discovery – specifically as a guide to the type of requirements generally involved in WS discovery. This leads to focussing on seven areas for WS discovery. In the conclusion we consider a mechanism to support the requirements raised in this paper.

1.1 Automated Web Service Discovery

First we interpret some terminology to facilitate later description: we use the term 'application' to mean software that has been developed to use other software applications (usually) owned by another party. We use 'whole system' to be the executing combination of the application software with its selected web services and any software that sits in between (e.g. needed to find or adapt to a WS).

Web services proponents use them in three scenarios: the first involves the application developer identifying a WS prior to designing their application, and developing it with the identified WS in mind. This predetermination of the service completely restricts the whole system to using that service – no automated discovery is involved. In the second scenario an application developer chooses a set of several services from which one will be selected as needed – possibly depending on availability. The choices are made before the (client) application software has been fully developed, but the scenario involves at least a degree of dynamic service selection during execution. However, no automated discovery takes place as the whole system executes. Both the first and second scenarios are now commonplace and are used in situations where system developers or integrators use web services as remote modules, which could be developed in-house, and are so under the control or influence of the system developer.

The third scenario is where the application is designed without any prior selection of a set of web services, such that the system discovers and selects from any that meets its requirements during execution, essentially at the point when the system needs it. This third scenario is what we believe quintessentially describes how web services should be used, but it is the most difficult to achieve. In fact it is rare – and it wholly depends on effective automated WS discovery.

1.2 Technical WS Discovery Solutions

The goal of automated web service discovery is that the client software finds and selects the required web services during execution to make up the whole system; there is no human intervention at that point. In principle the whole system may behave differently on each execution, as it discovers and uses different web services.

The current solutions for WS discovery are partial, often because they do not distinguish between the three scenarios above. They appear not to consider the business ideas that underpin web services, particularly to do with ownership and control: they are purely technical solutions. Such solutions are not a sufficient basis for automated WS discovery.

1.3 Using a Business Service Metaphor

Our stance in this paper is that the weaknesses of the solutions so far are due to a lack of consideration to all that is required of a web-based business service developed independently by others. Since this is a situation routinely faced in business, and since the ideas of web services and SoA are supposed to reflect business, we propose to seek solutions for fully automated WS discovery by using business practices for service discovery and selection as a model.

We next examine the problems with current solutions before proceeding to characterize how businesses locate and choose services which are needed for their business function, but which they do not or cannot supply for themselves, sometimes for financial reasons.

2 Current WS Discovery Solutions

Due to space limitations in the following brief review, we focus on a selection of recent contributions. For a review of older contributions see [2].

2.1 UDDI Enhancements

Because it appears to dominate perceptions about WS discovery, despite its well-documented deficiencies [3] we start with work on UDDI. It is important here to mention the use of the UDDI and in particular the work of Ran [4]. We argue that the UDDI does not allow for quality (what Ran calls quality of service) metrics nor does it have the capability to include semantic information. Ran proposes extending the UDDI. This is despite exposing the following weaknesses:

1. The UDDI is largely unregulated
2. The links in the UDDI are unstable
3. The entries are only functional, with the non-functional (i.e. performance) attributes omitted.

These three points, with the earlier two, are enough to suggest that, if we are to match business processes successfully, the UDDI should be replaced and not extended. It is essential that we have some mechanism that enables dynamic discovery and selection. Our proposed business service selection model aims to support these five requirements that have been found lacking in the UDDI.

Despite these failings, the UDDI is still being used. Aiming for efficiency, Lee *et al.* built a mechanism for service retrieval that was "implemented on a relational DB and cooperated with a UDDI registry" [5]. Their solution serves to highlight the problems inherent in the UDDI and as such will not succeed without taking into account other factors/problems such as the semantics and the WSDL descriptions that we discuss later.

Alrifai has proposed an "architecture" called iConnect that uses WSs to distribute data and how it supports "instant availability" [6]. This solution is also limited in that it too uses the UDDI. In particular it does not use any semantic data, and it relies on the links in the UDDI being accurate. iConnect also relies on the WSDL file for a particular web service; the problem with this is that this file is deficient because it

does not describe all the information needed for successful discovery. For example, the standard WSDL file lacks semantic descriptions. "iConnect" also needs human intervention to find a suitable web service.

Chen & Abhari [7] propose a new "framework" for dynamic service selection. However, this framework relies solely on the UDDI and as such encounters the problems we have previously discussed. As well as the fact that this framework is not yet implemented by Chen & Abhari we infer that not all the problems with web service discovery are addressed with this framework.

2.2 Other Solutions

Moving away from the UDDI, there are other attempted solutions to automate web service discovery. Rouached & Godart propose a "run-time service discovery process for web service compositions" [8]. The main weakness of this solution is that it disregards any web service that fails to meet precisely the application's functional and non-functional needs of a service. However, this approach does nicely encapsulate the requirement that a discovered web service can be adapted for use by software.

Ma, *et al.* [9] concentrate on semantics, emphasizing the view that irrelevant web services can be eliminated from search results based on the data the service returns. We infer from this and other similar work, a requirement that there is a need for a much fuller description. Otherwise, web services are disregarded only because the available description was not accurate enough. Thus a situation can arise in which a WS actually does what is required but would not be used because its description was lacking.

2.3 Distributed or Centralized Discovery Solutions

In this paper we have already considered some of the work related to the topic of WS discovery. In this section we will look at the important subjects of centralization, the UDDI and semantic information.

One issue for such WS discovery requirements is to do with centralization: must some middleware be interposed between requesting software and a WS, or must a peer-to-peer relationship between the two be maintained at all costs? Banaei-Kashani developed a discovery "architecture" that was based on peer-to-peer nodes: "WSPDS ... is a decentralized discovery service with peer-to-peer architecture for the Web Services infrastructure" [10].

This decentralized approach leaves itself open to misuse and brings about reduced service support. With a centralized discovery mechanism (whether called an "architecture", "framework", or "infrastructure") there is an opportunity to build confidence in service discovery because there is a record of web services and how they have performed. "Good practice" requirements come as a result. A common problem for business is that business people need to see some proof that a service is of value. A centralized approach can bring with it the means to provide for a community memory of web services and a community influence on those who provide them. Some centralized pooling of knowledge addresses many of the problems raised in previous sections of this paper.

The lack of business confidence is a common barrier to adoption of web services: "Current testing techniques for web services are unable to assure the desired level of trustworthiness, which presents a barrier to WS applications in mission and business critical environments" [11]. Taking a decentralized approach does not remove this barrier as there is a lack of metadata about a particular web service. In a centralized approach, with each component corresponding to a business practice, we can have different perspectives of use. With each perspective we have the opportunity to fine tune the discovery process. A decentralized approach does not assign ownership. Therefore, there is no business user available to be responsible for the service provided. This does not match business functions where some entity takes responsibility for the services it provides. A centralized approach aims to ensure consumer confidence.

2.4 Semantic Concerns

Regarding semantic information, Rajasekaran presents "an approach, which allows software developers to incorporate semantic descriptions of Web services during code development" [12]. We suggest that this is not the best point to add semantic information to a web service: since we are discussing ontologies it would be more accurate to include semantic information at a higher level. We further suggest that the semantic information should be added when the candidate web service subscribes to the directory. This will take the responsibility away from the programmer and give it to a higher-level business process. In addition, this would give the semantic information a different perspective than that provided by the programmer of web services. The programmer does not necessarily have access to the higher-level business process so does not see the relevant semantic information.

We have investigated these recent and other (older) WS discovery solutions and our primary, general conclusion is that they fail to look at all the requirements for WS discovery. Our secondary observation is that even in the specific areas that have been focused on in these solutions there are failings, which we have highlighted. (The adherence to the UDDI mechanism is a problem.) Finally, we note that these (and other) discovery solutions do not look at the problem from the viewpoint of WSs matching business practices.

3 A Business Model for Web Service Discovery

If web services represent business processes and services in software, we argue that reasoning about business practices is more likely to help software-based practices for web services. To start with, web services are created by one company and made available to others under certain conditions, normally for financial profit. Therefore potential or actual consumers of a service do not directly control the functions or performance characteristics of a service they might use. If, as we propose, the software protocols for (software) web services need to some extent match those for (human) business services. To begin to articulate requirements for mechanisms to support automated WS discovery, we articulate practices in general business service identification and selection.

3.1 Generic Service Discovery/Selection

The process of a business finding a service to outsource some of its work to is routine, but complex; the discovery and selection process will involve, at a high level of abstraction, something like the following:

1. articulating as precisely as possible what is needed from the 'outsourced' service by the consuming business;
2. locating possible services and their descriptions of what is offered;
3. checking on the precise meaning of advertised services,;
4. obtaining knowledge about the quality of a service via referrals and endorsements via human networks of service users;
5. short-listing potential services for selection;
6. checking that the external service and the internal processes can be made to fit and negotiating adjustments when needed;
7. determining and/or negotiating prices and estimating overall costs of service use;
8. making final selection – possibly as a primary choice with one or two backups, each with likely differences in prices and process fits.

By comparison, for the software involved in a business to find, and then determine the suitability of the encapsulated software of another business is equally complex, and not at all routine (yet). Rather than devise piecemeal solutions to automated WS discovery, we propose to look for holistic solutions.

3.2 Software Service Discovery

We argue that the whole system – the requesting software (the client software), the web service, and any mediating software – needs to match (but not mimic) the discovery and selection processes that humans carry out in business. With this aim in mind, and after having undertaken a review of past and present WS solutions, the problems inherent in the area indicate that any solution needs to include a set of interacting mediating components. By highlighting the problems, we offer the research community an analysis of the field that potentially indicates several courses of action, i.e. several types of mechanism which can be implemented to support automated WS discovery.

As we shall discuss below, web services are not being developed for automated discovery. Applications are not being written to find and select, during execution, software that is offered as a service and that is written and controlled by others with whom the owners of the requesting software do not parley with in order to change the service.

Our starting position about service discovery and selection in service-oriented systems follows these general principles:

- Discovery of a service is not just about finding it in some sort of directory: people try to assess quality aspects by using referrals, asking for reference sites, looking for testimonials, etc.
- How a service is advertised may not convey adequately what using it means; detail is important in determining if a service provides what is needed – at least within acceptable parameters. In software this is about semantics.

- A candidate service may do what a business needs but requires that the client business interacts with it in a way that the client does not normally operate: some special procedure that adapts a client's operation to that which the service needs may be required. In software this is about syntax and adapters.
- If a client business needs to pass some of its assets (or its customers' assets) to the service, it needs to trust the service to look after those assets and not to compromise them in any way. In software this is about security.
- Over a period a client business may gather information on how well a service is meeting its needs so as to develop knowledge for the time when reassessment of service offerings is needed. In software this is represented by business intelligence.
- It may be in the interests of a client's business for the results of the first execution of a service to be available the next time the service is run, even if the service does not store such results.
- Many services are subject to guidelines, regulation or standards, although these can be used to obfuscate as well as elucidate; a client business may look for a reliable set of standards that properly constrain a service. In software we need to validate against standards for which compliance is claimed.
- During the course of finding and selecting a service, a business will look at whatever descriptions of the candidates it can find, whether or not the descriptions were furnished by the service provider; in many circumstances an expert will be consulted to see if he or she knows about apparently comparable services that have been rejected. In software this points to types of metadata and tools to analyze and support the metadata.

If web services correspond to business services and as such are to be the elements of a service-oriented system, software developers need a solution that enables their software to make software- and business-related decisions, and do the business- and software-based negotiations that a human might do now. This is an area that has been largely ignored because, we judge from the literature [13,14] that proponents of discoverable web services have not thought enough about the business analogy nor truly addressed the fact that a web service, just like a service in business, is outside the client's control. The lack of business focus has resulted in impoverished analysis of the requirements. We are looking at the requirements for web service discovery with the aim of dealing with both software and business decisions that need to happen. In this paper we articulate the requirements that will be needed to support web services discovery.

4 The Business of WS Discovery Requirements

Adopting a business-oriented stance, and with the business-related insights from technical solutions, we can now examine requirements that could form a complete set for supporting automated WS discovery.

4.1 General Requirements

Although the WS discovery requirements we are considering are expressed at a high level of abstraction, it is important here to consider some detail because of

interactions among requirements resulting from wide distribution of problems involved in WS discovery. A WS is conceptually different from other encapsulations of business behaviour in software because it is outside the control of the requesting software, and, its behaviour cannot be changed by negotiation as it is generally when one business wants to utilize software originating in another. As Turner et al. [1] put it regarding SaaS, there is a separation of "the *possession* and *ownership* of software from its *use*".

There are technical and business reasons why we would want a system in which requesting software could find and use services without human intervention and across business boundaries. From the business viewpoint it would shorten the supply chains and enhance business–to–business cooperation. It is envisaged that software could fully assess a potential service without human intervention – resulting in fewer delays while humans (inconsistently) assess candidate services in software. Automated WS discovery should result in more consistent business decisions; however, flexibility and adaptability are more highly valued in business over consistency, so those attributes must be realized by any WS discovery mechanism.

A major concern is for requesting application software to 'tell the world' that it needs a service. In general, this is not yet an automated task. The present situation is that the web service provider, in effect, says "this is what we do, do you want to invoke us?" This is done via the WSDL file. What is needed is a mechanism whereby the web service client says "this is what I need, who can service me?" Then we need to match the requesting software to the potential web services for subsequent use. According to a business model, what needs to happen next is that the list of candidate web services needs to be whittled down to those that are somehow most suitable. Part of the business-style suitability check is that standards requirements be considered: the WSDL files of the candidate web services would need to be checked to see if it complies with the W3C standard [15]. All these tasks are not automated; it would need the software developer to work manually through these steps.

4.2 Semantic Information

Web services are short on semantic information and as a result "pervasive networked devices and programs that can seamlessly interoperate are still a way off." [16] Nayak states that "Due to the lack of semantic descriptions of the Web services, the search results returned by the service registries are effectively inadequate." [17] This results in potential mismatches between requesting software and potential web service providers. A web service needs to bring with it a semantic description that can be interrogated by requesting software. These semantic descriptions are needed to develop domain knowledge about the web services and the requesting software. If two pieces of software have a well-defined semantic description it will be easier to make them interoperable. This will ensure that from a business standpoint the requesting software and the candidate WS are semantically matched. This does not happen at present.

4.3 Adapters

The next issue we have identified is in the use of syntactic adapters [18]. Again, without human intervention, it is possible that there will be instances when some

requesting software will not interoperate with a web service because of a technical mismatch. From a business stance the syntactic adapters can cover instances where two parties with different forms need to interact. This is a common business problem where two organizations want to do business together but there is a point of contact that is not readily compatible. What happens now is that the software developers of the client-side have to negotiate with the software developers of the server-side to ensure that two pieces of software interoperate. The mismatches between service interfaces and protocols are hard to pinpoint [19]. This is a must-have requirement for WS discovery.

In order for a web service to support different interfaces and protocols there is a need to provide multiple interfaces [20]. At present the developers of the requesting client software are having to design and implement the interfaces that they need to interoperate with a particular web service.

4.4 Standards Compliance

WS discovery is affected by the use of standards or interpretations of standards. It is, from a business perspective, unhelpful to discover or select a web service that does not comply with application-required standards, or claims incorrectly to comply with required standards, or complies with ineffective standards. The client needs some form of assurance. It is up to the WS provider to offer evidence of conformance, or it is possible that the provider could sub-contract this process to another party.

The standards area relating to web services and SoA is unsatisfactory. There are too many standards organizations [21] all competing for the same ground, with each organization having its own set of beliefs and viewpoints. What consequently happens is that when a choice needs to be made about which WS to use the decision is biased because a particular viewpoint is favoured when the choice is made about which standard to comply with. The business client considers the influences brought to bear on any provider of a service. The same must happen when choosing which web service to use. This approach matches the selection process followed from a business perspective.

With proprietary interest influencing standards for automated WS discovery, interoperability diminishes. In the business service selection model we propose that proprietary interests are negated as far as possible so that the following considerations are addressed: financial impact of employing a proprietary standard; a limited pool of knowledge if a particular proprietary standard is used; limit the effect of in-house commercial strategies driving standards down one track and not the other; bypass disputes concerning IPR; and avoid infrastructure limitations caused by implementing a particular proprietary standard. These business factors will all assist in automating WS discovery.

Another issue with standards is that they are seen as not being constraining enough. What would happen if a doctor prescribed a medicine that was unlicensed in a particular country? Due to the unknown side effects the patient may suffer, a third party regulatory body would be notified, would investigate independently to see if the doctor behaved irresponsibly and possibly take action. In this example, the laws are not strict enough to prevent this situation happening again (it is beyond the scope of this paper to discuss other ethical and medical reasons why this situation may occur). The same must happen from a standard's perspective. For example, Section 7 of the SOAP

specification does not prevent software users from ignoring levels of trust. For the client this could have considerable impact on their business practices. This example from SOAP is a circular argument in that technically a business process must be implemented. If, for instance, Business A trusted Business B with the names and addresses of customers and, Business C obtained access to this data because the level of trust was ignored, then the results of giving away sensitive commercial data could be severe. Therefore, using the proposed business service selection model as a guide, we recognise requirements to tighten loopholes in standards that are used. The application (software) will have the option of asking for a service that complies with "Standard X" as it is, or, whether they want a third party to intervene and police "X", thus guaranteeing its constraints. In our example, this would mean strengthening Section 7 of the SOAP specification; for a definition of differences between standards, specifications, etc. see [22].

From a business perspective, the policing of standards will need to be considered as a requirement for automated WS discovery if the client application is to 'have faith' in the offered WS. What happens now is that a developer needs to spend resources making decisions about which, if any, standards to require of a potential WS. Therefore, it would be helpful if a business had the equivalent of a best practice document that leads them to the most appropriate standards body for their needs. The requirement then for the business service selection model is that it incorporates a mechanism that assesses the standards and categorises them according to their inherent features.

4.5 Security

We have already indicated in the previous section of this paper that there are problems with web service security. There are instances where the web service sessions are deemed insecure [23]. Namli and Dogac state that "… the privacy and security issues are indispensable for Web service technology in order to make them acceptable in more sensitive business transactions"[24]. If we want an automated web service discovery mechanism that matches business practice then we must address these security issues.

The requirements then for the security section of the business service selection model is that web services match the commercial, social and personal traits in a way that enhances web service interoperability via a discovery discourse. Our model takes a snapshot of the security needs of a client and then matches that to a web service. This matchmaking ensures that the expectations of the client are fulfilled so that we have secure sessions and levels of security that are acceptable to the client and the particular context in which they want to use the web service. For example, in our model there would be different levels of security according to the transaction type. If there is a financial aspect to the transaction, the level of security will be higher than if there were none. Our model addresses business-related security issues to ensure that the client is satisfied with the level of trust and security.

4.6 Business Environment Change

The next area of concern deals with changes in the business environment. In §3 of this paper we sketched the way businesses typically select a service. A business's

application software that uses another's web service will usually have to change if the prevailing business environment changes.

Automated discovery cannot be used in conventional web service collaborations that inherently have to deal with changes in the business environment. Even if the application software can adapt to a changing environment, people would have to be involved in selecting different web services for changes in functionality requirements. This would match the service selection procedures found in general business practice. There is a requirement then in our model for a mechanism that allows for changes in the business environment.

If web services can match the processes involved in coming to business-related decisions, the degree of interoperation increases, both in technical and business terms.

4.7 The Issue of State

In the business environment there is a concern about who owns and manages business objects that are used in the provision of products and services. If, for example, a business needs a service that retrieves publications based on a title and author search "looses" the specific title then all the publications by that particular author are retrieved; the resultset is too large and is inaccurate. Who is responsible for ensuring that the title of the publication is present in the search term? Is it up to the business that is requesting the service or do the providers of the service have to take responsibility for ensuring they have all the required data? We believe that there needs to be a third-party that manages the data on behalf of the service user and service provider.

State management is an issue for discovery of web services in that flexible and efficient procedures needs to manage state on both sides of the web service environment [25]. By this we mean the memory that a requesting application needs before and after using a service. In automated WS discovery there is a requirement therefore for a mechanism that manages the memory state of data.

The issue of state is about the ownership of data and who takes responsibility for ensuring its validity. From a business point of view will enterprises be comfortable with letting unidentified users (requesting software) access this data? Numerous matters need to be considered when answering this question. Some like standards and security we have investigated in this paper. Others like social and political factors are beyond the scope of this paper. Our investigations lead us to believe that the issue of state management is a requirement in our business service selection model.

4.8 Metadata

Before a business will use a service provided by another business it takes into consideration many factors. These factors contribute to a large proportion of the decision-making processes. By this we mean the reasons why a business will use a particular service provider. For example, the length of time it takes to provide a service, or the reputation ranking of a particular service. In the business sense it is therefore a requirement to have data about the data used in a service.

To improve automated WS discovery it is an analogous requirement to have a repository of the metadata about web services; not just data about its operations and interfaces but a record of how it has performed. With this record, web services could

be highlighted as "high performers". In our business service selection model it is a requirement that we hold structured data about WSs. This data would describe the characteristics of a particular WS. We would use this data the next time a web service becomes a candidate for use. For example, we would measure how long it took a particular WS operation to complete on a given day at a given time. With this execution rate recorded in a structure imposed by the business service selection model the next time the WS is considered for use it is possible to gauge whether it is fast enough. This speed of execution could also have implications for cost. That is one measure in our business service selection model that drives interoperability.

5 Conclusions

Web services are, and were always meant to be, a software encapsulation of business processes and, as such, they offer certain benefits, like interoperation with other software and software-based service discovery. The paradigm both implicitly and explicitly supports the possibility of businesses that have developed processes in software to offer them as third-party suppliers to other businesses for use by their software-based processes. Unfortunately, the full set of business benefits are yet to be realized because WS discovery is not fully automated in practice. What we have observed in WS discovery solutions to date are what might be termed technical solutions in which idealized simplifications of finding and using services have been used. There has been a real absence of representations of business practice in solutions that try to support the discovery and selection of web services from third parties.

The main aim of this paper was to articulate requirements for a solution to the problems surrounding web service discovery by matching the general practices of business in discovering and utilizing services. We went back to the original idea of web services – to have a mechanism for software applications to utilize functionality for business processes provided by others.

We identified the requirements for service discovery by undertaking a literature review (part of which was included here). In this review we focused on what was hindering service discovery. This review brought to the fore requirements concerning semantic information, syntactic adapters, standards, security, business intelligence, data management and support services. Requirements for supporting interoperability also emerged.

The results of our research are the requirements that we have explored throughout this paper. We argue that it is important to address these points as a whole, not in isolation. There does not appear to be a solution to automated WS discovery that addresses these requirements as a whole.

Regarding future work: we are currently implementing a software mechanism, named the Web Service Architecture, which supports the requirements above. This work so far is showing that the benefits derived from the set of requirements are possible given certain design choices.

Our discussion in this paper has articulated the need to consider the whole set of requirements; requirements that must be interpreted according to a business sense. It is not sufficient to look merely for partial, technical fixes.

References

1. Turner, M., Budgen, D., Brereton, P.: Turning Software into a Service. Computer 0018-9162/03, 38–44 (2003)
2. Fustos, J.: Web Services–A Buzz Word with Potentials. In: USDA Forest Service Proceedings, vol. 2006 (2006)
3. Al Masri, E., Mahmoud, Q.H.: Investigating web services on the world wide web. In: WWW 2008: Proceeding of the 17th international conference on World Wide Web, pp. 795–804 (2008)
4. Ran, S.: A model for web services discovery with QoS. ACM SIGecom Exchanges 4(1), 1–10 (2003)
5. Lee, K.H., Lee, K.C., Kim, K.O.: An efficient approach for service retrieval. In: Proceedings of the 2nd international conference on Ubiquitous information management and communication, pp. 459–464. ACM Press, New York (2008)
6. Alrifai, R.: An architecture that incorporates web services to support distributed data. Journal of Computing Sciences in Colleges 23(4), 241–246 (2008)
7. Chen, Y., Abhari, A.: An agent-based framework for dynamic web service selection. In: Proceedings of the 2008 Spring simulation multiconference (2008)
8. Rouache, M., Godart, C.: A run-time service discovery process for web services compositions. In: Proceedings of the 10th international conference on Electronic commerce. ACM Press, New York (2008)
9. Ma, J., Zhang, Y., He, J.: Efficiently finding web services using a clustering semantic approach. In: Proceedings of the 2008 international workshop on Context enabled source and service selection, integration and adaptation: organized with the 17th International World Wide Web Conference (WWW 2008). ACM Press, New York (2008)
10. Banaei-Kashani, F., Chen, C.C., Shahabi, C.: WSPDS: Web Services Peer-to-Peer Discovery Service. In: Intl. Symposium on Web Services and Applications (2004)
11. Tsai, W.T., Wei, X., Chen, Y., Xiao, B., Paul, R., Huang, H.: Developing and assuring trustworthy Web services. In: Proceedings of Autonomous Decentralized Systems, 2005. ISADS 2005, pp. 43–50 (2005)
12. Rajasekaran, P., Miller, J., Verma, K., Sheth, A.P.: Enhancing web services description and discovery to facilitate composition. In: Cardoso, J., Sheth, A. (eds.) SWSWPC 2004. LNCS, vol. 3387, pp. 55–68. Springer, Heidelberg (2005)
13. Vitvar, T., Mocan, A., Kerrigan, M., Zaremba, M., Zaremba, M., Moran, M., Cimpian, E., Haselwanter, T., Fensel, D.: Semantically-enabled service oriented architecture: concepts, technology and application. Service Oriented Computing and Applications 1(2), 129–154 (2007)
14. Cauvet, C., Guzelian, G.: Business Process Modeling: A Service-Oriented Approach. In: Proceedings of the Proceedings of the 41st Annual Hawaii International Conference on System Sciences (HICSS 2008) (2008)
15. W3C. Web Services Description Language (WSDL) 1.1. W3C (2001), `http://www.w3.org/TR/wsdl` (accessed: 16-2-2008)
16. McIlraith, S., Martin, D.L.: Bringing Semantics to Web Services. In: IEEE Intelligent Systems, January/February 2003, vol. 1094-7167/03, pp. 90–93 (2003)
17. Nayak, R., Lee, B.: Web Service Discovery with additional Semantics and Clustering. In: Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence, pp. 555–558. IEEE Computer Society Press, Washington (2007)

18. Lahiri, T., Woodman, M.: Web Service Architectures Need Constraining Standards: An Agenda for Developing Systems without Client-Side Software Adapters. In: Proceedings of the IASTED International Conference on Software Engineering, pp. 45–52 (February 2006)
19. Nezhad, H.R.M., Benatallah, B., Martens, A., Curbera, F., Casati, F.: Semi-automated adaptation of service interactions. In: Proceedings of the 16th international conference on World Wide Web, pp. 993–1002. ACM Press, New York (2007)
20. Benatallah, B., Casati, F., Grigori, D., Nezhad, H.R.M., Limos, I., Campus des Cezeaux, B.P., Aubiere Cedex, F.: Developing Adapters for Web Services Integration. In: Pastor, Ó., Falcão e Cunha, J. (eds.) CAiSE 2005. LNCS, vol. 3520, pp. 415–429. Springer, Heidelberg (2005)
21. Bell, G.: A Time and a Place for Standards. Queue 2(6), 66–74 (2004)
22. Lahiri, T., Woodman, M.: Web Service Standards: Do We Need Them? In: Pautasso, C., Bussler, C. (eds.) Emerging Web Services Technology. Springer/Birkhauser (2007)
23. Bhargavan, K., Fournet, C., Gordon, A.D., Corin, R.: Secure Sessions for Web Services. ACM Transactions on Information and System Security 10(2) (Article 8) (2007)
24. Namli, T., Dogac, A.: Using SAML and XACML for Web Service Security and Privacy. Securing Web Services: Practical Usage of Standards and Specifications (2007)
25. Song, X., Jeong, N., Hutto, P.W., Ramachandran, U., Rehg, J.M.: State Management in Web Services. In: IEEE Proceedings of the 10th IEEE International Workshop on Future Trends of Distributed Computing Systems (FTDCS 2004) [0-7695-2118-5/04], pp. 1–7 (2004)

Part V

Human-Computer Interaction

An Agile Process Model for Inclusive Software Development

Rodrigo Bonacin[1], Maria Cecília Calani Baranauskas[2], and Marcos Antônio Rodrigues[1]

[1] CTI, MCT, Rod. Dom Pedro I, km 143, 6, 13082-120, Campinas, SP, Brazil
{rodrigo.bonacin, marcos.rodrigues}@cti.gov.br
[2] Institute of Computing, University of Campinas, UNICAMP
Caixa Postal 6176, 13083 970, Campinas, Brazil
cecilia@ic.unicamp.br

Abstract. The Internet represents a new dimension for software development. It can be understood as an opportunity to develop systems to promote social inclusion and citizenship. These systems impose a singular way for developing software, where accessibility and usability are key requirements. This paper proposes a process model for agile software development, which takes into account these requirements. This method brings together multidisciplinary practices coming from Participatory Design, and Organizational Semiotics with concepts of agile models. The paper presents the instantiation of the process model during the development of a social network system, which aims to promote the social and digital inclusion. The results and the adjustments of the proposed development process model are also discussed.

Keywords: Accessibility, Agile Methods, Organizational Semiotics.

1 Introduction

The increase in software complexity and consequently development costs, as well as the demand for quality and productivity resulted in the need for organizing the software development tasks. Many software engineering research projects attempt to establish process models to make the software development a more predictable and productive task. Some process models are very systematic, and advocate performing many extra activities in addition to the core development ones. These activities usually produce a lot of documentation and demand a lot of resources.

Another problem of many processes is the difficulty to deal with context changes during the software development. The agile methods [1], aim to be flexible to deal with these changes, focusing on the core development activities. The quality and productivity are achieved by focusing on the individuals, by working most of the time with the software itself, by collaborating with customers, and by being agile enough to respond to changes.

Nowadays, the Internet represents a new dimension to the software development, where systems, people, and business can globally communicate. According to the World Wide Web Consortium [21], the social value of the Web is that it enables human communication, commerce, and opportunities to share knowledge. These

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 807–818, 2009.
© Springer-Verlag Berlin Heidelberg 2009

benefits should be available to all people, independently of their hardware, software, network infrastructure, native language, culture, geographical location, physical or mental ability. These aspects are related to both social and technological issues. However, the Internet is not accessible by everyone, especially in the development countries with many illiterate people.

Producing systems that are accessible by everyone represent a big challenge to the Human-Computer Interaction field. The objective of this work is to search for alternatives to deal with accessibility and usability issues considering the quality and productivity of software development. The proposed method relies on bringing together multidisciplinary practices coming from Participatory Design (PD), and Organizational Semiotics (OS) with concepts of agile development models.

This model was instantiated for the development of a social network system, which aims to promote the social and digital inclusion. This social network is part of the project entitled: “e-Cidadania: Systems and Methods for the Constitution of a Culture mediated by Information and Communication Technology" [4]. The project investigates and proposes solutions for the diversity of users and competencies that constitute the scenario of the digitally excluded people in the Brazilian society (which include illiterate and impaired people). To reach this goal, the research group develops joint actions with a partner institution (network Jovem.com and communities around it) to conduct interaction and interface design of a pilot system to support inclusive social networks, to be implemented in the target community.

The paper discusses difficulties and adjustments in the model occurred during the development of this system, including potential challenges in the use of agile methods in research projects involving academic development teams.

The paper is organized as follows: Section 2 presents some key concepts of the agile models and methods, PD and OS; Section 3 shows the rationale behind the conception of the proposed model, Section 4 describes it; Section 5 presents and discusses the instantiation of the model, and Section 6 concludes the paper.

2 Background

In this section we detail the main background work used to delineate the proposed process model. Section 2.1 introduces the agile methods, the main values, principles and practices. Section 2.2 presents the Human-Computer Interaction foundations and the Participatory Design. Section 2.3 presents Organizational Semiotics practices, which are used in a participatory way in the proposed process model.

2.1 The Agile Methods

The term “Agile Method” is used to identify a set of methods which follow common agreements to respond to changes during the software development projects. This term date from 2001, when leading proponents of “light” methodologies wrote the agile manifest [1], which states:

“We are uncovering better ways of developing
software by doing it and helping others do it.
Through this work we have come to value:

Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan
That is, while there is value in the items on
the right, we value the items on the left more."

They also agreed in eleven agile principles, which give more detail and more concrete meaning to these values. With the popularization of the methods, the agile alliance [2], a Non-Profit Organization, was created to support agile software projects and help people start new projects. This alliance is formed by several institutional and individual members from the industry and academy spread out in the world. Its website became probably the main source of information about the agile methods, including hundreds of articles, events information, books, programs, and resources in general.

Nowadays, there are many agile methods. Although the agile methods follow the same basic principles, they have expressive differences. Koch [8] presents a systematic way to evaluate which method is more adequate to a specific organization. In his approach, evaluators rank features of each method in worksheets concerning the organizational context and the agile method values. The evaluation is summarized and the results point out the organizational changes and the appropriated methods to be applied. Koch [8] considers some of the most popular methods:

- **Adaptive Software Development (ASD).** This method views a software project team as a complex adaptive system that consists of agents, environments, and emergent outcomes. It is based on the following cycle: Speculate, Collaborate, and Learn;
- **Dynamic Systems Development Method (DSDM).** This method leaves the details of software writing relatively undefined and instead focuses on system development. The most important part is not the process flow itself but the set of nine principles on which that process was built: active user involvement is imperative; the teams must be empowered to make decisions, the focus is on frequent delivery of products, fitness to business purpose is the essential criterion for acceptance of deliverables. Iterative and incremental development is necessary to converge on an accurate business solution, all changes during development are reversible, requirements are baselined at a high level, testing is integrated throughout the life cycle, and a collaborative and cooperative approach between all stakeholders is essential;
- **eXtreme Programming (XP).** It is a widely recognized method, and adopts the practices, which became reference to many agile projects. The adopted practices are the planning game, small releases, use of metaphor, simple design, test first, refactoring, pair programming, collective code ownership, continuous integration, work 40-hour week, on-site customer, and use of coding standards;
- **Feature-Driven Development (FDD).** FDD differs from other agile methods in its focus on upfront design and planning. FDD is designed by eight practices: domain object modeling, developing by feature, Class (code) ownership, feature teams, inspections regular build schedule, configuration management, and reporting/visibility of results;

- **Lean Software Development (LD).** It is not really a method; LD can be understood as a set of principles and tools that an organization can employ in making its software development projects "more lean". The principles of LD are: eliminate waste, amplify learning, decide as late as possible, deliver as fast as possible, empower the team, build integrity in, and see the whole. These principles are supported by 22 tools, based on lean production concepts;
- **Scrum.** It is a method for managing product development that can be wrapped around any specific technology, including software. The scrum practices includes: the scrum master, product backlog, scrum teams, daily scrum meetings, sprint planning meeting, sprint, and sprint review.

2.2 Human-Computer Interaction and Participatory Design

Researchers and practitioners in the HCI field have developed and assessed various methods and tools for the promotion of quality in use of interactive systems. Aspects related to human factors in the use of these systems have been given special emphasis. In order to refer to the design of systems that are allied with the user's point of view and expectations, Norman and Draper [14] introduced the term "User-Centered Design" (UCD). Nowadays, this is a very active line of research in the HCI field.

Almost twenty years after the introduction of the "User-Centered Design", Norman [15] argues that this approach has a limited view of design. He argues that the design process must also consider all human activities in an activity-centered view. Another alternative for user-centered design approaches is usage-centered design, proposed by Constantine et al [5]. His approach, which focuses on the task instead of the user, involves successive refinements of models (e.g. user role model, user task model, and interface content model) to fit the final user interface design.

A strategy for achieving a broader view of design is to encourage the potential users to participate during the whole engineering life cycle. By using this approach, the interface is designed with the users, so that they can participate in design decisions, and express their opinions about activities, practices, tasks and usage context. This participatory approach in system development has its roots in work done in Scandinavia during the seventies and promotes direct user involvement in various phases of the design process, including problem identification and clarification, establishment of requirements and analysis, as well as high level design, detailed design, evaluation, user customization and redesign [13]. In PD, collaboration between designers and users is considered essential to achieve democracy in decision making, quality in use and acceptance of the product. PD also promotes mutual learning [7] from the combination of different experiences.

A large amount of research in the PD field has been conducted to establish meaningful practices for the provision of a common ground for discussion among those directly involved in the design and use of the technology [18]. Participatory techniques are useful instruments for the discussion of the social context of users, since they promote active participation. PD researchers have developed techniques that explore different approaches to promote productive worker-designer co-operation. These techniques have the aim of providing designers and workers with a way of connecting current and future work practices with envisioned new technologies. Participatory design techniques do not suppose to be a straightforward sequence of

well-understood steps that produce a guaranteed outcome, but a scaffold or an infrastructure for a complex group process.

Nevertheless, as argued by Gronbaek et al [6], PD techniques seldom go beyond the early analysis/design activities of project development. Pekkola et al. [16] argue that a multi-methodological approach using prototyping and a set of various means of communication is necessary to stimulate end-user participation in development processes. It is important that the people involved share a model for the representation of the work domain which will be involved in the prospective system.

Meaning-making is constructed as a result of cooperation among designers, developers, interested parties and prospective users of the technology being designed. Therefore extensions and adaptations of the techniques are necessary for inclusive environments (e.g. use of tactile cues, use of high contrast solutions, sign language interpretation, and assistive technology compatibility) to enable the participation of people with different needs, skills and interests, including people with disabilities [11]. The participation of each and every individual must be promoted so that he/she can perceive, understand, communicate, be understood and interact with his/her peers [10].

2.3 Participatory Practices Based on Organizational Semiotics

Semiotics has been an area actively attended to by scientists in linguistics, media studies, educational science, anthropology, and philosophy, to name a few. Computing, including the development and use of computer systems, is another field in which Semiotics has shown a great relevance.

Organisational Semiotics (OS), a branch of Semiotics which understands the whole organization as a semiotic system [9], aims at studying organizations using concepts and methods based on the Semiotics developed by Peirce [17] and Morris [12]. OS understands that each organized behavior is affected by the communication and interpretation of signs by people, individually or in groups. As a "doctrine of signs", Semiotics encompasses several disciplines, facilitating our understanding about the ways people use signs for every type of purposes. Organizational Semiotics presents theories and methods to allow the analysis and design of information systems in terms of three human information functions: expressing meanings, communicating intentions and creating knowledge [20]. In the philosophical stance underlying OS, reality is seen as a social construction based on the behavior of agents participating in it; people share a pattern of behavior governed by a system of signs.

OS Methods have been found useful for dealing with the influence of social aspects in organizations and in the elicitation of system requirements. Among the methods employed by the OS community is a set of methods known as MEASUR (Methods for Eliciting, Analyzing and Specifying Users' Requirements) [19], which deal with the use of signs, their function in communicating meanings and intentions, and their social consequences. MEASUR involves the analysis of stakeholders in a focal problem, as well as considering their needs and intentions, and the constraints and limitations related to the prospective software system.

In the process model presented in this paper we propose "participatory workshops" inspired by and conducted using Organizational Semiotics artifacts to organize the discussion and support the meaning making process. The OS diagrams are made into large posters and hung on the wall to mediate discussions during the workshops.

A facilitator introduces the discussion based on the Semiotic artifacts. In a brainstorming format, the participants express their ideas using post-its that are stuck on the posters. During brainstorming the discussion includes the position of the post-its on the diagram, so that related ideas are grouped together constituting a certain structure. The participants are also invited to discuss conflicting ideas, even though some conflicts will only be resolved during later stages of the project. After the brainstorming, the participants are invited to synthesize the themes discussed, and these are then presented to all.

3 Delineating a Development Process

The initial proposal of the method was to establish a systematic way to develop the system proposed in the "e-Cidadania" project [4]. From the specific process requirements for this project the process model was generalized to deal with accessibility and usability issues.

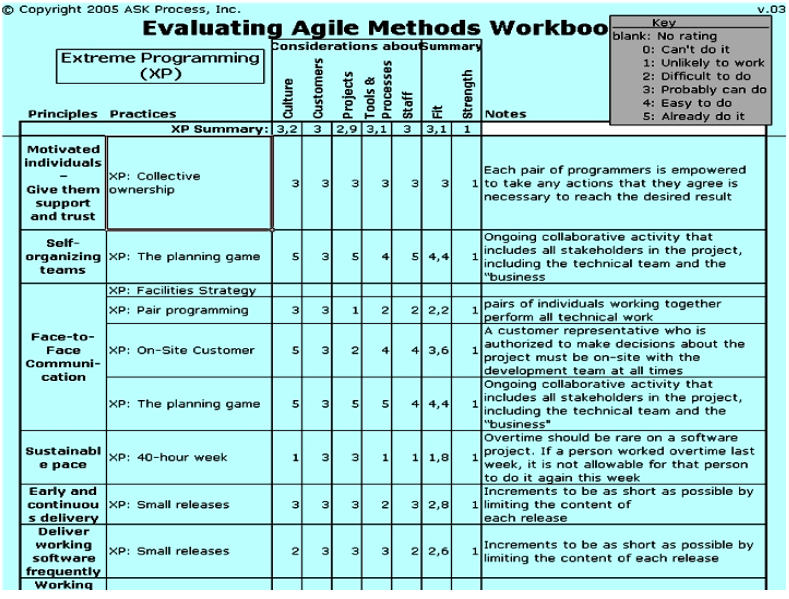
The first step conducted to delineate the model was to analyze the agile principles in order to verify the compatibility of these principles with our development context. The main aspects of the "e-Cidadania" context are:

- “e-Cidadania” is a research project that includes development activities;
- The team is composed of researchers, students, and other professional contracted to codify the system;
- The team is geographically distributed;
- The software is innovative in terms of the accessibility and usability of its user interface;
- The development team follows a flexible timescale.

The next step involved an analysis of the most popular methods, using the Koch [8] approach. The intention was to identify process features that could be interesting to our context. These were considered in the construction of the proposed process model. Figure 1 shows an example of part of the worksheet constructed to evaluate principles and practices of the XP method.

The methods practices and principles were discussed in a first meeting with the developers, project researchers and representatives of the target audience. From the discussion, we have extracted some examples of issues to be addressed and possible solutions, for example:

- Problem: "We have distributed teams, resulting in difficulties to practice pair programming and daily meetings". Discussed solution: To investigate the use of collaborative development tools and to use practices adopted by the "open source" development communities;
- Problem: “We need models”. Discussed solution: Investigate the use of agile modeling [3];
- Problem: "Accessibility and usability are key requirements". Discussed solution: promote the user participation with PD methods; adopt the W3C standards, use low-fidelity prototype, etc.



© Copyright 2005 ASK Process, Inc. v.03

Evaluating Agile Methods Workboo

Extreme Programming (XP)

Considerations about | Summary

Principles	Practices	Culture	Customers	Projects	Tools & Processes	Staff	Fit	Strength	Notes
	XP Summary:	3,2	3	2,9	3,1	3	3,1	1	
Motivated individuals – Give them support and trust	XP: Collective ownership	3	3	3	3	3	3	1	Each pair of programmers is empowered to take any actions that they agree is necessary to reach the desired result
Self-organizing teams	XP: The planning game	5	3	5	4	5	4,4	1	Ongoing collaborative activity that includes all stakeholders in the project, including the technical team and the "business
Face-to-Face Communi-cation	XP: Facilities Strategy								
	XP: Pair programming	3	3	1	2	2	2,2	1	pairs of individuals working together perform all technical work
	XP: On-Site Customer	5	3	2	4	4	3,6	1	A customer representative who is authorized to make decisions about the project must be on-site with the development team at all times
	XP: The planning game	5	3	5	5	4	4,4	1	Ongoing collaborative activity that includes all stakeholders in the project, including the technical team and the "business"
Sustainable pace	XP: 40-hour week	1	3	3	1	1	1,8	1	Overtime should be rare on a software project. If a person worked overtime last week, it is not allowable for that person to do it again this week
Early and continuous delivery	XP: Small releases	3	3	3	2	3	2,8	1	Increments to be as short as possible by limiting the content of each release
Deliver working software frequently	XP: Small releases	2	3	3	3	2	2,6	1	Increments to be as short as possible by limiting the content of each release
Working									

Key
blank: No rating
0: Can't do it
1: Unlikely to work
2: Difficult to do
3: Probably can do
4: Easy to do
5: Already do it

Fig. 1. Example of Method Evaluation

After initial meetings, the process principles were synthesized. From these principles a lifecycle model and practices were defined. The principles, lifecycle and practices were discussed with developers, researchers and representatives of the target users in another meeting. The process model was also revised by specialists in accessibility and usability, and finally the model was instantiated in the e-Cidadania project context to carry a case study.

4 The Agile Inclusive Process Model

The Agile Inclusive Process Model (AIPM) follows principles relatively distinct from other agile process models. While the other processes focus mainly on the production of software and its quality, the AIPM focus on accessibility and usability of the final product. The AIPM principles are described as follows:

- Promote the participation of the users and other stakeholders' with the universal access and inclusive design values;
- Construct a shared vision of the social context;
- Include more than just technical issues in the development of the system;
- Promote the digital inclusion through participatory activities.

In order to develop the software in an agile way, the general principles and values of the agile alliance are also considered in addition to the AIPM principles. Figure 2 presents an overview of the AIPM lifecycle. This lifecycle follows the XP idea of adopting nested lifecycles.

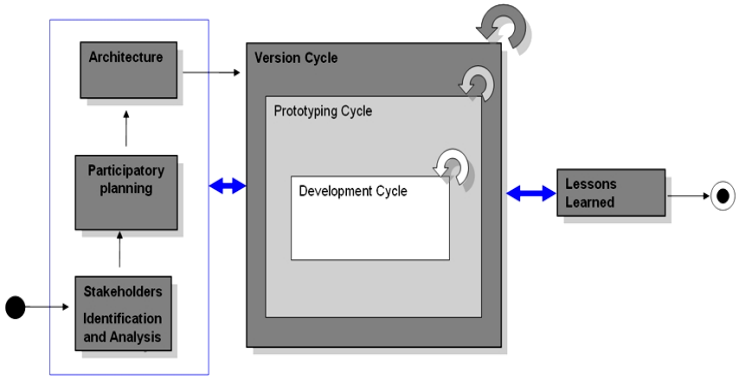


Fig. 2. An Overview of AIPM Lifecycle

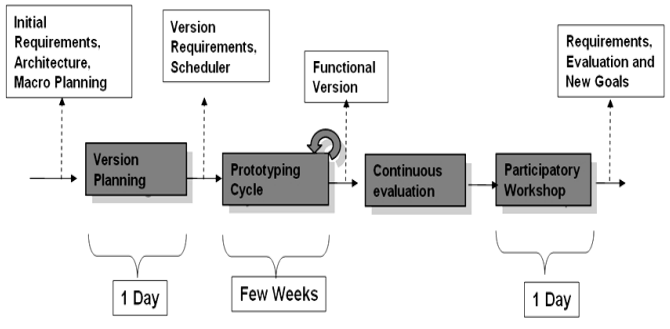


Fig. 3. Version Cycle Details

As Figure 2 shows, the first practice is the "stakeholder identification and analysis", where the designers, developers, specialists on digital inclusion, and community members participate in workshops to discuss the possible stakeholders and the project context. The next steps are the "participatory planning" and the definition of the "architecture". After few days, the version cycle is initiated, but these three practices still occur in parallel till the end of the project. Each version cycle takes no more than one or two months and involves 3 or 4 prototyping cycles. Each prototype cycle takes three to four weeks involving 3 or 4 development cycles that takes around seven days each. According to Figure 2 the lessons learned are also synthesized in parallel during all the lifecycle.

Figure 3 presents details of the version cycle; the grey boxes represent practices and activities, the upper boxes the main artifacts consumed or produced, and the boxes below, the time required for each grey box. This cycle starts with the "Version Planning" which delineates the main version requirements and a macro schedule for the prototyping cycle.

After the first prototypes a "continuous evaluation" is conducted, and at the end of the cycle a "Participatory Workshop" is performed. OS methods and tools support these workshops.

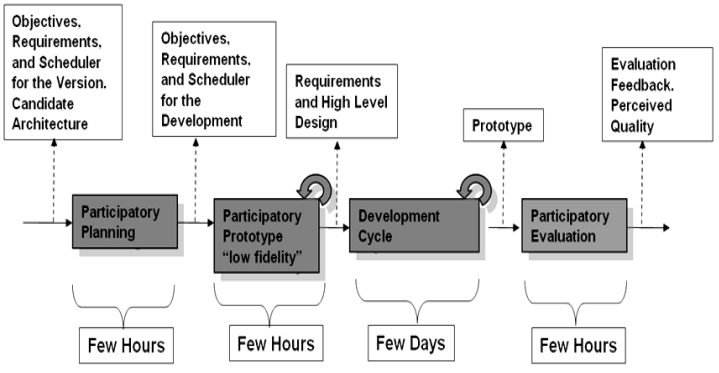


Fig. 4. Prototype Cycle Details

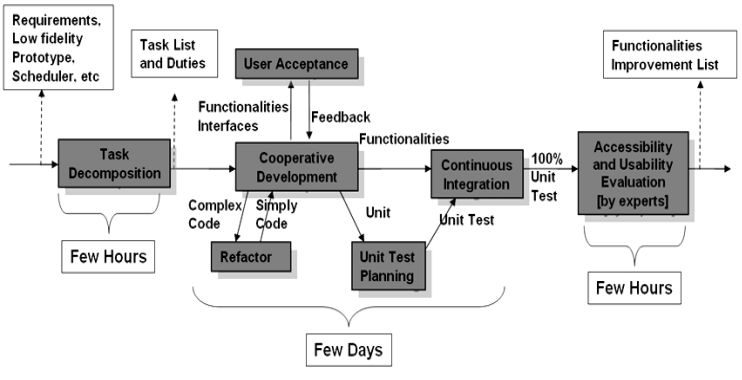


Fig. 5. Development Cycle Details

Figure 4 presents the details of the prototyping cycle. This cycle starts with the "participatory planning" for the prototype and development. The "Participatory prototype" practice is conducted in order to define the requirements and the high level design of the interface using "low fidelity" prototypes. After that, the prototype is implemented during the development cycle and evaluated using PD techniques.

The development cycle (Figure 5) is based on XPs "Refactor and Continuous Integration" practices, but the proposed cycle also aims at producing accessible and inclusive software. The "task decomposition" is the first step proposed; it is a meeting between the developers and designers to identify tasks and to attribute duties over each task. During the "cooperative development" the functionalities, navigation models, screens, and other elements of the interface are implemented using tools for supporting the cooperative development. The new functionalities and interfaces are frequently evaluated by the users, as well as the code frequently refactored and continually integrated. When all the unit text is completed the code is inspected by accessibility and usability experts.

5 The AIPM in Practice

In this section we present the design and development activities conducted during the construction of the first and second prototyping cycles of the eCidadania software, named *Vilanarede*.

As Figure 6 shows the software construction follows the AIPM lifecycle. We started the development by the early august, and each development cycle took around one week (usually in middle of the week). By the middle of September the first increment was deployed, and the second by the middle of October. These increments were only accessible for a selected set of users, providing the continuous feedback for the design and development team. A first Beta Version was deployed at an Internet server, for the whole community without access restrictions by the end of the year. For the first Beta Version of the system two main functionalities of the social network were selected: (1) user access control, including registration and login, and (2) announcement edition and publication.

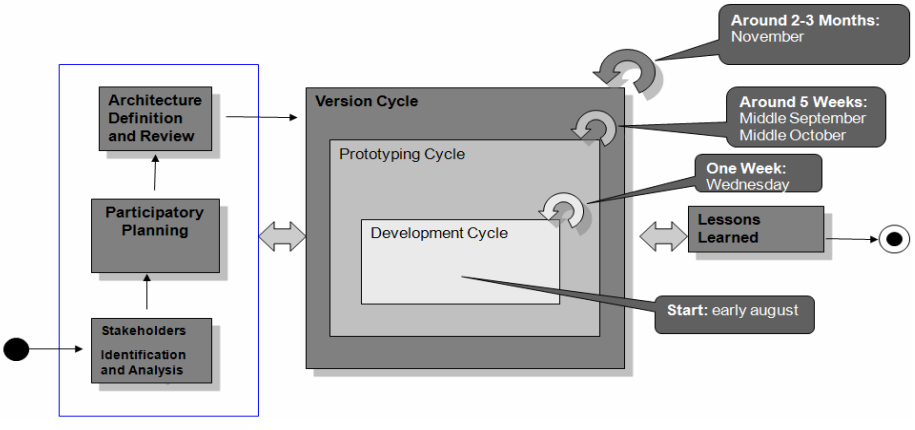


Fig. 6. An Overview of AIPM Lifecycle Instantiation

From this experience, we can highlight and discuss some difficulties and the lessons learned:

- **Environment Setup Problems.** The use of proper development tools was an essential requirement to be agile, especially in distributed teams. However, setting up the environment takes time. The development velocity was tremendously increased after fixing environment problems. Aspects such as: control version, development tools compatibilities, communication tools, and application server setup had influenced directly the development. Based on our experience, we can say that it is better to spend some resources preparing the environment and fixing it to be agile in the future, even if this task result in an apparent losing of time;
- **Face-to-face Communication Still Important.** Although the use of communication artifacts such as email and on-line messages can help in minimizing the interaction problems, the face-to-face meetings are still necessary in the project. Some

misconceptions regarding aspects of the user interface design were removed faster during the meetings;

- **Each Activity has its Appropriate Time.** Agile cannot be a synonymous of "Quick-and-dirty" development. This is especially important when we are talking about the accessibility and usability activities. The AIPM includes many activities related to these issues, including experts analysis, low fidelity prototyping, user acceptance analysis, and workshop with the users. We cannot do it fast at any cost; firstly it is necessary do it with the required quality. Adapting these practices to be executed in an agile manner respecting the quality is a challenge that we have to deal with.
- **An Academic Development Team has Different Time Constraints.** When adapting an agile method to a development team which is not in a software development organization, other interests and activities compete for the time. For example, within the research team, development was not a full time activity of the team; coding must be balanced with other tasks such as participating in the workshops with users, analyzing data, writing papers, etc. It has been a challenge to get a balance while maintaining the deadlines established in the version planning, for high level prototyping and coding.

6 Conclusions

The Internet represents a new dimension to software development. It can be understood as an opportunity to develop systems that support the social inclusion and citizenship. However, those systems impose a singular way for developing software, in which accessibility and usability are the key requirements.

This paper presented a process model, based on agile methods, as well as practices, methods and theories from Human-Computer Interaction, Participatory Design and Organizational Semiotics. The process model was defined to meet conditions and requirements of the e-Cidadania project based on principles, a lifecycle model, and fundamental practices. The process model is now in use. Although many aspects of a legitimate agile process model had to be adapted to the context of development we had in the Project, the overall process produced good results, and each proposed activity, mainly the related to accessibility issues, is being improved to be more agile.

Acknowledgements. This work is funded by Microsoft Research - FAPESP Institute for IT Research (proc. n. 2007/54564-1).

References

1. Agile Alliance: Manifesto for Agile Software Development, `http://agilemanifesto.org/` (accessed February 12, 2008)
2. Agile Alliance: How The Agile Alliance Operates, `http://www.agilealliance.org/show/1646` (accessed February 12, 2008)
3. Ambler, S.W.: The Object Primer: Agile Model-Driven Development with UML 2, 3rd edn. Cambridge University Press, Cambridge (2004)

4. Baranauskas, M.C.C.: e-Cidadania: Systems and Methods for the Constitution of a Culture mediated by Information and Communication Technology. Proposal for the Microsoft Research-FAPESP Institute (2007)
5. Constantine, L., Biddle, R., Noble, J.: Usage-Centered Design and Software Engineering: Models for Integration. In: IFIP Working Group 2.7/13.4, ICSE 2003 Workshop on Bridging the Gap Between Software Engineering and Human-Computer Interaction, Portland, Oregon, USA, pp. 3–10 (2003)
6. Gronbaek, K., Kyng, M., Mogensen, P.: Toward a Cooperative Experimental System Development Approach. In: Kyng, M., Mathiassen, L. (eds.) Computers and Design in Context. MIT Press, Boston (1997)
7. Kyng, M.: Designing for Cooperation: Cooperating in Design. Communications of the ACM (1991) Doi: 10.1145/125319.125323
8. Koch, A.S.: Agile Software Development: Evaluating the Methods for Your Organization. Artech House, Norwood (2005)
9. Liu, K.: Semiotics in information systems engineering. Cambridge University Press, Cambridge (2000)
10. Melo, A.M.: Design Inclusivo de sistemas de informao na web. PhD Thesis, University of Campinas, Campinas, Brazil (in Portuguese) (2007)
11. Melo, A.M.: Baranauskas MCC An inclusive approach to cooperative evaluation of the web user interfaces. In: Proceedings of the 8th International Conference on Enterprise Information Systems, Paphos, Cyprus, pp. 23–27 (2006)
12. Morris, C.W.: Foundations of the theory of signs. International Encyclopedia of Unified Science 1(2) (1938)
13. Muller, M.J., Haslwanter, J.H., Dayton, T.: Participatory Practices in the Software Lifecycle. In: Helander, M.G., Landauer, T.K., Prabhu, P.V. (eds.) Handbook of Human-Computer Interaction, 2nd edn. North-Holland, Elsevier, Amsterdam (1997)
14. Norman, D.A., e Draper, S.W.: User centered system design: New perspectives on human-computer interaction. Lawrence Erlbaum, Hillsdale (1986)
15. Norman, D.A.: Human-Centered Design Considered Harmful. Interactions 12(4), 14–19 (2005)
16. Pekkola, S., Kaarilahti, N., Pohjola, P.: Towards Formalised EndUser Participation in Information Systems Development Process: Bridging the Gap between Participatory Design and ISD Methodologies. In: Proceedings of the 9th Participatory Design Conference, Trento, Italy, pp. 1–5 (2006)
17. Peirce, C.S.: Collected Papers. Harvard University Press, Cambridge (1958)
18. Schuler, D., Namioka, A.: Participatory design: principles and practices. Lawrence Erlbaum Associates, Mahwah (1993)
19. Stamper, R.K.: Social Norms in requirements analysis - an outline of MEASUR. In: Jirotka, M., Goguen, J., Bickerton, M. (eds.) Requirements Engineering, Technical and Social Aspects. Academic Press, New York (1993)
20. Stamper, R.K.: Organisational Semiotics: Informatics without the Computer? In: Liu, K., Clarke, R., Andersen, P.B., Stamper, R.K. (eds.) Information, Organisation and Technology: Studies in Organisational Semiotics. Kluwer Academic Publishers, Dordrecht (2001)
21. World Wide Web Consortium, `http://www.w3.org/` (accessed February 12, 2008)

Creation and Maintenance of Query Expansion Rules

Stefania Castellani, Aaron Kaplan, Frédéric Roulland, Jutta Willamowski, and Antonietta Grasso

Xerox Research Centre Europe
6 chemin de Maupertuis, 38000 Grenoble, France

Abstract. In an information retrieval system, a thesaurus can be used for query expansion, i.e. adding words to queries in order to improve recall. We propose a semi-automatic and interactive approach for the creation and maintenance of domain-specific thesauri for query expansion. Domain-specific thesauri are especially required in highly technical domains where the use of general thesauri for query expansion introduces more noise than useful results. Our semi-automatic approach to thesaurus creation constitutes a good compromise between fully manual approaches, which produce high-quality thesauri but at a prohibitively high cost, and fully automatic approaches, which are cheap but produce thesauri of limited quality. This article describes our approach and the architecture of the system implementing it, named Cannelle. It exploits user query logs and natural language processing to identify valuable synonymy candidates, and allows editors to interactively explore and validate these candidates in the context of a domain-specific searchable knowledge base. We evaluated the system in the domain of online troubleshooting, where the proposed method yielded an improvement in the quality of the search results obtained.

Keywords: Information retrieval, Query expansion, Domain-specific thesaurus, Knowledge base management.

1 Introduction

Domain-specific searchable knowledge bases (KBs) such as online troubleshooting search systems contain technical content describing for instance mechanical parts, operations thereon, configuration settings, etc. Naive users of such a system often have trouble searching it because they are unfamiliar with the domain-specific terminology, whether because it consists of technical words they do not understand, or simply because there are many possible terms for the same thing—for example, the place in a photocopier where the blank paper is loaded might be called a drawer, a tray, or a feeder. To bridge this terminology gap, it can be helpful for an IR system to apply query expansion rules that automatically add additional terms to user queries, so that for example a query containing the word "drawer" will be matched with results containing the word "feeder".

To be useful, query expansion rules must be chosen carefully. A generic list of synonyms such as a writer's thesaurus typically is missing many domain-specific synonymy relations, and at the same time includes synonymies that are inappropriate in specific contexts in the KB. Query expansion rules can be more effective if they are tailored

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 819–830, 2009.
© Springer-Verlag Berlin Heidelberg 2009

to the particular KB to which they will be applied. However, it can be difficult for KB editors to write such rules by hand. The process involves first identifying candidate rules, e.g. by interviewing KB users about searches they have made that were unsuccessful, or by poring over query logs looking for queries that seem likely to have failed, or by pure intuition. Once a candidate rule has been identified, it must be evaluated to see whether it brings a decrease in precision that outweighs the increase in recall; this involves posing many different queries to the IR system and comparing the results.

In this paper we describe Cannelle, a tool that supports and partly automates the process of identifying and evaluating query expansion rules. We have developed and tested the tool in the context of an online troubleshooting KB for office devices such as printers, but we believe the approach would be applicable to domain-specific text collections of other kinds as well, and particularly ones that use standardized terminology with which users might not be familiar.

2 Related Work

We are not aware of any previous work on hand-construction of domain- or corpus-specific query expansion thesauri (We use the term "thesaurus" to mean a lexical resource used for query expansion, even though it may include not only synonyms as in a real thesaurus, but also terms related in other ways). There is, however, work on query expansion using hand-build general-purpose thesauri, and using thesauri constructed automatically from corpora (which can be domain-specific).

Query expansion using a general-purpose thesaurus such as WordNet [1] over a TREC corpus (non-domain-specific) can yield improved results for particular queries, particularly short queries, but can also significantly degrade results by adding contextually inappropriate words to the query [2]. In our own tests with a domain-specific KB and a general-purpose thesaurus, we found that relatively few queries were improved, since the most helpful synonymy relations were domain-specific ones that were absent from the thesaurus, but queries with degraded results were still common.

In [3], a general-purpose thesaurus is used in an information extraction context. Lexico-syntactic restrictions induced from example sentences in the thesaurus are used to disambiguate words to avoid contextually inappropriate application of synonymy rules. In an early version of our system we experimented with a similarly rich rule formalism, but for the moment we have abandoned this direction primarily because of interface problems: our approach involves machine-assisted development of rules by a person, and it proved difficult to design an interface with which a linguistically-naive user could write lexico-syntactic rule restrictions.

A number of techniques have been proposed that exploit statistics of term distribution in the text being searched in order to retrieve relevant documents that do not contain the original query words. This class of techniques includes latent semantic indexing [4], as well as query expansion using automatically-constructed thesauri such as presented in [5]. These techniques are based on the idea that if two terms tend to occur in similar contexts within the corpus being searched, then they are likely to be similar in meaning, and thus that a user who poses a query including one of them is likely to be interested in documents that contain the other. This kind of technique can only identify similarities

between two words both of which appear in the documents being searched. In the case of a curated technical KB, editors generally try to ensure that a given concept is referred to consistently using a single standard term, and thus the frequencies of non-standard synonyms of a standard term tend to be near zero. In other words, the terms for which we need synonyms are precisely terms that do not occur in the KB, so KB statistics are not helpful for finding synonyms. Applying similar techniques to a corpus other than the one to be searched, as in [6], might yield some useful synonym pairs, but then the non-corpus-specific nature of the resulting thesaurus would result in a loss of precision, as discussed above for general-purpose hand-built resources. For this reason, we use logs of past user sessions, rather than documents, as a source of statistics for identifying term similarities. The idea of using query logs as a source for query expansion has precedents in e.g. [7,8,9,10] and [11]. The details of the way in which we calculate term similarities from query log statistics differ from those of other systems, but we have not performed detailed comparisons and do not claim that our method is superior. The novelty of our system is in the way candidates are subsequently contextualized and evaluated using an interactive, semi-automatic tool.

Another approach to query expansion is to involve the user explicitly in selecting additional query terms, e.g. [12] and [13]. The studies reported in [14] examine how KB end users interactively exploit structured domain specific thesauri for query term selection and query reformulation. Our approach puts some of the burden of evaluating candidate expansion terms on the editor of the KB, rather than on the end user. This is feasible in the case of a domain-specific KB, where a small number of synonyms can have a positive effect on a relatively large number of searches. Note that the two approaches are complementary—synonymy rules defined by an editor can be applied automatically without user intervention, while other query refinement choices are left to the user.

3 Semi-automatic Thesaurus Creation and Maintenance

In general creating a thesaurus involves first identifying a set of promising synonymy candidates, and then evaluating each candidate to decide if it should be used as-is, used with contextual restrictions, or discarded. In our context promising synonymy candidates are candidates mapping frequently used query terms to corresponding technical terms (i.e. terms actually present and indexed in the KB). Such candidates can be automatically identified through the analysis of query session logs and of the searchable KB. Nevertheless, automatically identified candidates might not be appropriate or be too general when used as such for query expansion and in consequence introduce noise into the search results.

Therefore, it might be necessary to restrict their application. To restrict the application of a candidate, both terms, the query term and the technical term, can be contextualized: the query term with respect to the query context in which it was used, and the technical term with respect to context in which it appears in the KB. Our system allows the KB editor to evaluate the impact that a (possibly contextualized) candidate would have on the search results.

Accordingly, the Cannelle system proposes the following interaction areas to its user (see Figure 1):

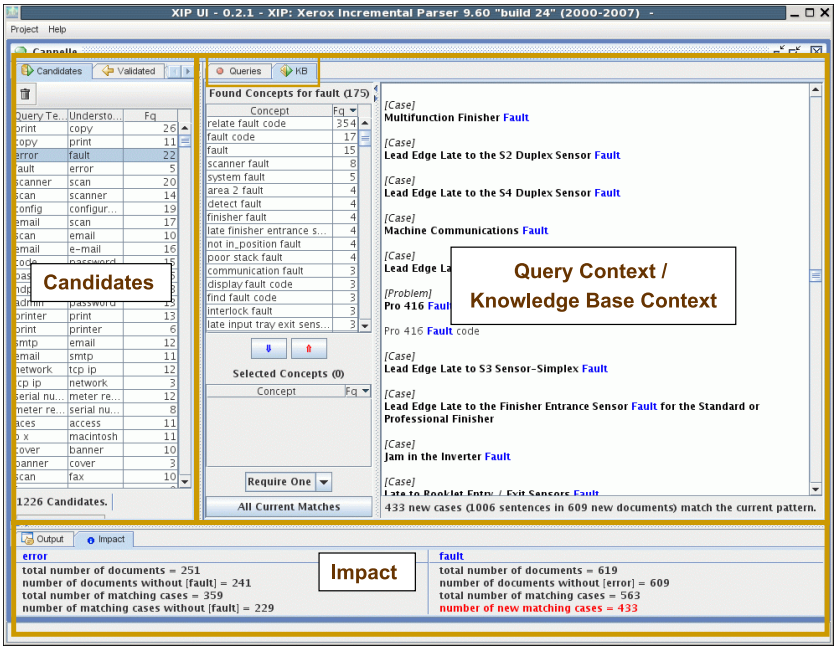


Fig. 1. Main interaction areas in Cannelle

- The list of synonymy candidates, i.e. pairs (query term, technical term), with the scores assigned to them by the selection heuristic (detailed below).
- The contexts in which the query / technical term appears in the query session logs / KB content respectively.
- The summary of the impact a synonym might have on user queries.

The synonymy candidates are displayed in the top left-hand area of the interface. The editor can select a candidate and evaluate it with respect to the impact it will have on typical user searches in the KB, in particular in terms of new results brought back through the introduction of the corresponding synonymy rule.

Figure 1 shows an example in the troubleshooting context, where the editor has selected the candidate (‘error’, ‘fault’) from the list of synonymy candidates. The editor can estimate if it is useful to create immediately a corresponding synonymy rule or not, or if a corresponding rule could be potentially useful but would introduce too much noise if introduced without restrictions. In the latter case, the editor can specify possible application contexts for the synonymy. These contexts are derived on one hand from the usage context of the query term in the session logs and on the other hand from the sentence contexts in the KB documents where the KB term appears. In our example (Figure 1), for the candidate (‘error’, ‘fault’) the editor may observe that ‘error’ is synonymous with ‘fault’ but only within sentences in the KB where ‘fault’ is part of the phrase “fault code”.

The system can be used by the KB editors at various stages. At the initial deployment of the KB, the system will provide support to the KB editors to evaluate opportunities

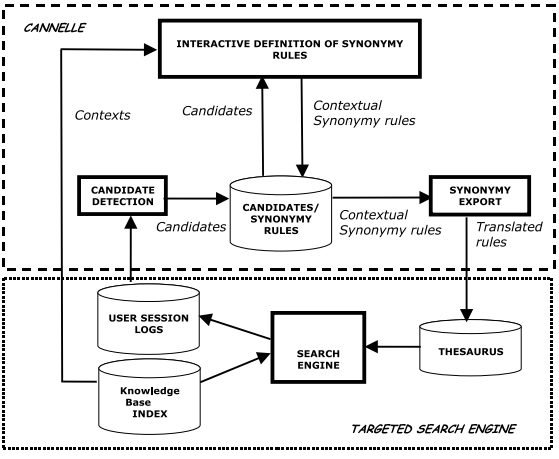


Fig. 2. Architecture of Cannelle

for the generation of new synonymy rules from other KBs system usage logs. When new content is created and added to the KB, the system can help the KB editor check if the new content also calls for modifications of the existing thesaurus, e.g. listing all the rules that apply to the new text. Then, periodically the system can be used to check if user terminology is still adequately supported by the KB. If not, or if a problem is detected, the system can help determine if adding new synonymy rules or modifying existing ones would help to better link unsupported user terminology with technical terms represented in the KB.

In the rest of this section we describe in more detail the various components and functionalities of the system starting with a description of its architecture.

3.1 Architecture

The main modules composing the system and the dependencies with the modules of the targeted search engine are shown in Figure 2.

A first module ("Candidate detection") identifies a list of candidates based on a heuristic which will be discussed below. These candidates are stored internally in the system and can be accessed by the other modules in further steps or updated with a subsequent run of the candidate detection module on new user session logs. The second module ("Interactive definition of synonymy rules") provides a graphical user interface in order to define some synonymy rules from the collected candidates. It collects from the KB sentences and syntactic contexts where a candidate will potentially impact a search. This information is presented to the editor who can then generate a contextual synonymy rule from a candidate. This rule is stored internally in the system. Finally, the module ("Synonymy export") is required to export the generated rules to the targeted search engine thesaurus, i.e. to translate them into the corresponding format.

3.2 Detection of Candidates for Synonymy

One can imagine different heuristics for identifying candidate synonymy pairs. Each heuristic detects candidate synonymy pairs and assigns a quality score to each candidate; if multiple heuristics are used, then the candidates can be ranked according to a weighted sum of the scores assigned by the individual heuristics. When the candidates are presented to the editor, those falling below a threshold score are automatically eliminated, and the score is used to rank those above the threshold.

For the current prototype of Cannelle we have defined and implemented a heuristic based on logs of past users' interactions with the KB. It requires that the logs contain information on the interactions of the user with the search engine during a session, i.e. that queries issued by the same user within a short period of time are grouped together. We assumed as elsewhere in the literature [7] that all queries in a session refer to the same problem and that such sessions have a first query formulation that has not brought satisfactory results. This first formulation is then followed by reformulations of the same problem description that are made by users in the attempt of finding what they are looking for.

Using this information, we count reformulation frequencies: for each pair of terms (X, Y), we count how often a query containing X is followed in the same session by another query that is identical except that X is replaced by Y. Each term X and Y can be composed of one or more words. For example, if a user issues the query 'error code' and subsequently the query 'fault code,' we count this as one occurrence of the reformulation error → fault. If another user makes the query 'scanner error' and subsequently 'scanner fault,' it is counted as another occurrence of the same reformulation. The reformulation pairs are taken as synonymy candidates, and their reformulation frequencies are used as scores. We filter out reformulation pairs whose replacement term does not occur in the KB, since using such pairs as query expansion synonyms would have no effect on search results.

In our experiments, many of the reformulations ranked highly by this heuristic are corrections of spelling errors, e.g. ('configuation', 'configuration'). If the query interface already includes a spelling corrector, then adding misspellings to the thesaurus would be redundant. We thus filter respellings as follows: we apply our spell checker to the problematic term in each candidate pair, and if it proposes the replacement term as a respelling, then we drop the pair from the candidate list.

3.3 Interactive Definition of Synonymy Rule

During the processing of a synonymy candidate (T1, T2) selected from the list of candidates, the editor can estimate that:

- it is not useful to create a corresponding synonymy rule, for example because it would introduce too much noise, or that
- it is useful to create a simple synonymy rule stating that any query containing T1 should be matched with all results containing T2, or that
- it is useful to create a synonymy rule but with contextual restrictions; then the work consists of exploring the contexts in which T2 appears in the KB, and possibly results in a synonymy rule specifying in which contexts the rule should apply.

In the first case, the editor can move the synonymy candidate (T1, T2) to a list of rejected candidates. In the second case, the editor can ask the system to directly create the synonymy rule for (T1, T2). In the last case, the evaluation allows the editor to explore the possible contexts of application of the synonymy. These contexts are automatically derived from the sentences in the KB documents. For example, for the candidate ('error', 'fault') the editor may observe that 'error' is synonymous with 'fault' but only within sentences where 'fault' is associated with 'code'. Choosing some contexts corresponds to constraining the synonymy rule so that the rule will be applied only within those contexts. Constraints can be applied to the context of the problematic term, the replacing term, or both.

Analysing the Candidates for Synonymy. The synonym candidates can be analysed with respect to the impact of the corresponding synonymy rules when searching the KB both from a quantitative and a qualitative point of view.

A first element of consideration for a candidate for synonymy (T1, T2) is the frequency of the replacement T1 → T2 in the queries. Also, for a given candidate for synonymy (T1, T2) the reverse pair (T2, T1), in particular if detected in the reformulations, can be taken into consideration at the same time. The editor can also see (1) the queries where a reformulation T1 → T2 has taken place, (2) the queries that contain T1 but which have not been reformulated, and (3) the set of sentences that would be retrieved from the KB for queries containing T1 if the synonymy rule for T1 → T2 were activated. Occurrences of the replacing term are highlighted in the sentences.

Some quantitative measures on the occurrences of the problematic term and replacing terms in the KB provide further support to evaluating the impact of introducing a synonymy rule:

- Number of documents where the problematic term occurs
- Number of documents where the replacing term occurs
- Number of sentences containing each of the terms
- Number of sentences containing the problematic term but not the replacing term and vice versa.

Specifying Contextual Constraints on the Problematic Term in Queries. From the comparison of the two lists of queries where the problematic term of a candidate appears, i.e. the list of queries that have been reformulated, and the list of non-reformulated queries, the editor can decide that the problematic term should be made more specific. For example, with the candidate ('code', 'password'), the queries in which 'code' was not reformulated as 'password' may contain 'area code' or 'fault code' whereas the reformulated queries may contain 'user code' or 'admin code'. A better candidate to consider for query expansion would be in this case ('user code', 'password') or ('admin code', 'password').

The editor can select a new problematic term from the list of reformulated queries or manually enter a new term. This will result in the creation of a new candidate in the system that can be processed in the further steps described below similarly than the ones that were automatically detected.

Specifying Contextual Constraints on the Replacement Term in the Knowledge Base. Ideally, if editors had unlimited time and patience, they would specify the list of KB documents that are relevant to each possible query term. Since it is typically not feasible for the editors to consider each document individually, we provide an interface that groups documents according to the context of occurrence of a replacement term. Editors can then choose contexts in which the rule should apply, and this has the effect of specifying entire groups of documents as relevant to the problematic term.

We use as contexts the syntactically coherent expressions identified by the method described in [13]. This method uses a parser to segment a text into a sequence of expressions. The granularity of the segmentation is defined such that expressions are typically quite short (a few words), so that there is a high probability of the same expression being found in multiple documents, yet long enough that each expression makes sense as a choice in a query refinement process. For example, the sentence fragment "image area partially blank when printing and copying" is segmented as follows: image area/partially blank/when printing and copying. Some normalization (stop word removal and lemmatization) is applied to increase the frequency with which equivalent expressions are found in multiple documents.

When the editor wishes to specify that a rule applies only when the replacement term is found in certain contexts in the KB, we propose as contexts all expressions (as defined above) that occur in the KB and contain the replacement term. For example, consider the pair ('handler', 'feeder'). The replacement term 'feeder' may be found in the expressions "automatic document feeder" (the part of the copier that handles stacks of originals) and "high capacity feeder" (a tray that holds blank paper). The editor can specify that queries for 'handler' should be matched with occurrences of "automatic document feeder", but not "high capacity feeder".

In the current version of Cannelle, the contexts are presented as a flat list, but a future version of the system will use a collapsible tree organized by subsumption. For example, Figure 3 shows the two alternatives to represent the contexts found for the term 'feeder' in our KB.

In the tree representation, selecting 'document feeder' would have the effect of selecting all documents containing either "document feeder" as a self-contained expression or "automatic document feeder"; the context 'automatic document feeder' would still be available as a choice, but as a refinement of 'document feeder' rather than as a separate choice.

Considering Symmetric and Transitive Rules. When evaluating a candidate for synonymy (T1, T2), the editor may be interested in considering the reverse pair (T2, T1) as well, for example because he knows that the two expressions are truly equivalent. In the

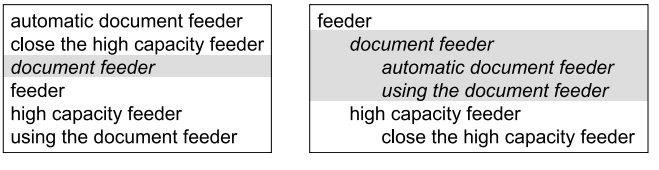


Fig. 3. Examples of context presentations (flat list and tree)

current version of the system, for each candidate for synonymy Cannelle just displays the reverse pair if the heuristics find evidence for it. In the next version of the system, after a rule has been created from a candidate, Cannelle will generate the list of additional candidates that can exist by symmetry. These are, for a rule where a term T1 will match T2 in the contexts C1T2 and C2T2, the new candidate pairs (T2, T1), (C1T2, T1), and (C2T2, T1). For example, if for the candidate ('smtp', 'email') the editor generates the rule 'smtp' → 'email server' — 'email setup' then the system will propose the following additional candidates: ('email', 'smtp'), ('email server', 'smtp') and ('email setup', 'smtp').

Another capability we are developing in Cannelle is the ability to generate additional candidate pairs considering the potential transitivity between synonymy rules. Following the same example used for the symmetry, if the rule 'smtp' → 'email server' | 'email setup' already exists and the editor generates the additional rule 'email' → 'e-mail' the system will propose by transitivity the following candidates: ('smtp', 'e-mail server') and ('smtp', 'e-mail setup').

4 Experiment

We conducted an experiment in order to estimate the capability of Cannelle to generate synonymy rules that improve the quality of the documents retrieved by a search engine. The experiment was designed in collaboration with the editors of the troubleshooting KB we have considered, i.e. the intended users of the tool. This experiment consisted in an evaluation of Cannelle using query logs from the KB collected over a year's time.

The evaluation consisted of the following steps:

1. Automatic identification of synonymy candidates from the user session logs;
2. Creation of a set of synonymy rules from the evaluation of the highest ranked candidates (including the reverse candidates);
3. Estimation of the impact of these synonymy rules in terms of number of sessions where they would be activated;
4. Estimation of the impact of these synonymy rules in terms of quality of results retrieved when rules are activated.

The list of the synonymy candidates was obtained by applying the heuristics on the user sessions covering the first 10 months (roughly 24 000 sessions, of which roughly 6 600 contained at least one query and roughly 2 000 contained reformulations). We then evaluated 80 candidates, consisting of the most frequent candidates and their reverses. The evaluation consisted in determining if these candidates constitute desirable query expansion rules or not, and if the rules needed to be restricted to certain contexts. Half of the synonymy candidates were evaluated by two different evaluators, the other half by only one evaluator. 57 of the candidates were approved. In addition, during the process the evaluators identified 3 additional synonymies that were not in the candidate list. Moreover, the evaluation of 3 candidates became redundant with respect to rules generated for previous candidates. In total this process lead to the specification of 60 synonymy rules, and more precisely of:

- 37 rules without context
- 23 rules with context

Table 1. Examples of synonymy candidates and corresponding generated rules

Problematic term → replacing term	Query context	KB context
email→e-mail		
error→fault		'fault' not followed by 'interrupted'
code→password	'admin code' or 'invalid code' or 'access code'	
sheet→page		'banner page' or 'cover page'
toner→cartridge		'cartridge' not preceded by 'staple'
printer→print	'printer device' or 'printer service' or 'printer driver'	'print device' or 'print service' or 'print driver'

- 3 only with query context
- 13 only with KB contexts
- 7 with both query and KB contexts

Table 1 shows some examples of synonymy candidates evaluated during the tests together with the query contexts and/or KB contexts, if any, in the corresponding synonymy rules generated.

In the final evaluation, we analyzed the impact of the generated synonymy rules on the last two months of logged user sessions. We counted in how many sessions the synonymy rules would have applied, and evaluated the quality of results returned with and without the rules. To evaluate the quality of the results we considered only the first query to which rules would apply within each session (later queries are less interesting, since the results returned for the first query would probably have influenced the user's subsequent queries). From these queries, 100 unique queries were randomly selected, and for each of them two different evaluators rated the relevance of the 20 top-ranked documents retrieved by the search engine with and without the synonymy rules. For each query, evaluators gave a score from 1 to 5, with 1 indicating that results with the synonyms were much less relevant than results without the synonyms, and 5 indicating the reverse. We then averaged the scores of the two evaluators for each query.

In our evaluation the specified synonymy rules applied in 38% of the sessions, which gives an indication of the significance of the method's scope. In 16% of queries the quality of the retrieved documents was improved by the application of the synonyms (score of 4 or above). We observed a decrease in result quality (score of 2 or below) for none of the tested queries, and the KB editors considered this an important outcome. Given the positive outcome of our evaluation, the editors have agreed to test the tool in their working environment. The tests are ongoing and will complement the results of our experiment, and they will also help in refining the next version of the tool, which we are currently developing.

5 Conclusions

We have developed a tool for interactive development and testing of query expansion rules. The editor of a domain-specific KB can use the tool to find

contextualized rules that are likely to improve the recall of many user queries while minimally affecting precision. The tool automatically proposes candidate rules using a heuristic based on logs of past query sessions. When the editor wishes to contextualize a rule, candidate contexts are suggested automatically based on query logs and on the application of natural language processing techniques to the KB text. Initial tests indicate that rules discovered using this tool significantly improve information retrieval results for a device troubleshooting KB, and the tool is currently being tested in a production environment.

We are developing a new version of the tool that will use a tree representation for presenting the contexts to the editor (see Section 3.3) and allow the evaluation of additional candidates for synonymy by symmetry and/or by transitivity (Section 3.3). This new version will also integrate the feedback we will receive from the tests in the production environment.

References

1. Fellbaum, C. (ed.): WordNet: An Electronic Lexical Database (Language, Speech, and Communication). The MIT Press, Cambridge (1998)
2. Voorhees, E.M.: Query expansion using lexical-semantic relations. In: SIGIR 1994: 17th ACM International Conference on Research and Development in Information Retrieval, pp. 61–69. Springer, New York (1994)
3. Jacquemin, B., Brun, C., Roux, C.: Enriching a text by semantic disambiguation for information extraction. In: LREC 2002: 3rd International Conference on Language Resources and Evaluation, pp. 45–51 (2002)
4. Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., Harshman, R.: Indexing by latent semantic analysis. Journal of the American Society for Information Science 41, 391–407 (1990)
5. Qiu, Y., Frei, H.P.: Concept based query expansion. In: SIGIR 1993: 16th ACM International Conference on Research and Development in Information Retrieval, pp. 160–169. ACM, New York (1993)
6. Turney, P.D.: Mining the web for synonyms: PMI-IR versus LSA on TOEFL. In: Flach, P.A., De Raedt, L. (eds.) ECML 2001. LNCS, vol. 2167, pp. 491–502. Springer, Heidelberg (2001)
7. Amitay, E., Darlow, A., Konopnicki, D., Weiss, U.: Queries as anchors: selection by association. In: HYPERTEXT 2005: Sixteenth ACM Conference on Hypertext and Hypermedia, pp. 193–201. ACM Press, New York (2005)
8. Baroni, M., Bisi, S.: Using cooccurence statistics and the web to discover synonyms in a technical language. In: LREC 2004: 4th International Conference on Language Resources and Evaluation (2004)
9. Cucerzan, S., Brill, E.: Extracting semantically related queries by exploiting user session information. Technical report, Microsoft Research (2005)
10. Jones, R., Rey, B., Madani, O., Greiner, W.: Generating query substitutions. In: WWW 2006: 15th International Conference on World Wide Web, pp. 387–396. ACM, New York (2006)
11. Cui, H., Wen, J.R., Nie, J.Y.: Query expansion by mining user logs. IEEE Trans. on Knowl. and Data Eng. 15, 829–839 (2003) (Member-Wei-Ying Ma)

12. Fonseca, B.M., Golgher, P., Pôssas, B., Ribeiro-Neto, B., Ziviani, N.: Concept-based interactive query expansion. In: CIKM 2005: 14th ACM International Conference on Information and Knowledge Management, pp. 696–703. ACM, New York (2005)
13. Roulland, F., Kaplan, A., Castellani, S., Grasso, A., Roux, C., O'Neill, J., Pettersson, K.: Query reformulation and refinement using nlp-based sentence clustering. In: Amati, G., Carpineto, C., Romano, G. (eds.) ECIR 2007. LNCS, vol. 4425, pp. 210–221. Springer, Heidelberg (2007)
14. Shiri, A.A., Revie, C., Chowdhury, G.: Thesaurus-assisted search term selection and query expansion: a review of user-centred studies. Knowledge Organization 29, 1–19 (2002)

Stories and Scenarios Working with Culture-Art and Design in a Cross-Cultural Context

Elizabeth Furtado, Albert Schilling, and Liadina Camargo

Universidade de Fortaleza – UNIFOR, CEP 60811-905, Fortaleza, CE, Brazil
elizabet@unifor.br, albertschilling@gmail.com, liadina@unifor.br

Abstract. This paper discusses the use of user experience prototyping and theatrical techniques in two experiments to attaint the following objectives of the interaction design: to explore new ideas and to communicate the cross-cultural users' needs and their expectations for iDTV (interactive Digital TeleVision) services. These two objectives are particularly important when are involved systems which are unknown to people. On the first experiment, we showed the implication of real stories for the construction of efficient interaction scenarios in a process of interaction design creation. On the second experiment, we showed the implication of stories told through theatre in order to have an objective communication of the purposes of iDTV services in a process of art and culture. The results are described by discussing the strengths and weaknesses of this approach.

Keywords: Experience prototyping, Theatrical technique, Stories, Scenarios, Interaction design.

1 Introduction

There is a range of works highlighting the importance of scenarios to describe the needs and objectives of users about a product yet to be developed [21]. We found also works describing the importance of stories to organize and transmit information, explore new ideas and communicate culture [11]. Both works refer to techniques of creating narrative sequences, telling what a person does (or should do), in which order, in which context, and what happened (or should happen) as a result of his/her actions [19]. To both techniques, storyboards are drawn as a way of representing graphically its information (such as the individuals, contexts, manipulated objects, etc.) [22].

In HCI, the boundary between scenario and story definitions is not clear. One of the greatest challenges is to understand how these techniques can be used together, particularly, how one can complement on the description and exploitation of the other. Hachos and Redish [12] discuss four levels of scenario: brief scenarios, vignettes, elaborated scenarios, and complete task scenarios. These scenarios grow from a brief description of a user's goals a product must handle, to a specific story a user trying to reach a goal but without details of steps, to one that shows the details of the tasks and steps in the interactions. This idea leads to move from initial discovery and exploration of the context to detailed interface and interaction design.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 831–842, 2009.
© Springer-Verlag Berlin Heidelberg 2009

Our experience reveals a good way for developers of a new technology, which has great influence on the life of interested users (as in the case of the iDTV), applying these techniques would be to think that the relation among both is not determined by a sequential process of description from generic scenarios (from the user's goals) to specific stories (describing the steps to achieve such results). In our approach the relation between scenarios and stories goes through a description and exploration among what's real and what's possible in an intercalated, complementary and flexible way. In this text, we don't intend to be exhausting in the description of factors that influence the detailing of a scenario and of a story, to the point to define if the same ones make a good story/scenario, if there is a correct detailing or where one finishes and the other begins.

In the present we analyze how scenarios and stories can make an impact on the exploration of new ideas and communication of interaction design of the iDTV applications. We conducted two experiments in which professionals: i) applied test scenarios as a resource of experience prototyping [3] in order to explore new ideas for iDTV applications, having as result real stories told by the study's participants. The test scenarios are a brief description of activities that the participant can do with similar iDTV prototypes. The stories constitute the reflexive register of happenings lived by the storyteller or by other participants. The real stories are based on facts that really happened, in which the storyteller uses his/her own experiences, feelings, values and attitudes as a starting point to communicate and send information [15] and; ii) used the theater as a ludic resource to communicate the interactive and cultural aspects contemplated on the project of the interaction of iDTV services. The stories told on the theater are (re)written having as starting point the interaction scenarios generated from the first experiment. The interaction scenarios describe the activities or human tasks, and allow the exploration and discussion of contexts, needs and requirements [17].

Next, we discuss about HCI literature to examine how scenarios and stories have been used to achieve the following objectives: explore new ideas and communicate cultural aspects.

2 Scenarios and Stories in the Interaction Design and Communication Process

The deployment of a new technology requires developers to have background knowledge about the needs of the users inclined towards this technology. These professionals apply various practices, such as: the identification of what users expect and the analysis of the viability of such expectations [6].

When we are talking about everyday products, as it is the Digital TV, which tends to benefit an entire community, a contextual understanding must be done. Ethnographic methods have been applied [13]. Users can use diaries to record the lived experience with the technology [9], [16]. Professionals have also used scenarios [5] and personas [7] in an integrated way [1]. Stories about personas [19] associate to storyboards [8] can be used as resources in participatory designs. Role-playing techniques can be useful in experience prototyping design sessions in order to express better the particularity of the interaction in action and to communicate culture. In our

understanding, the communication of culture is art, erudition and other manifestation more sophisticated of the intellect and of human sensibilities considered collectively [2]. The theatre is a way representing the real world and, through this representation, the individual reacts, and experiments her/his own emotions, among other feelings. Storytelling in the theatre brings emphasis on the ludic and derives from the arts. The theatrical art is configured as emotional communication, and it provides a symbolic transformation of the world, making the universe more meaningful and ordained [4]. We wish to use theatre in order to have a greater impact on professionals by providing them a wider characterization about the possible iDTV scenarios and the target users.

3 Research Factors

We will explore both the implication of real stories for the construction of interaction scenarios and the stories told by the theatre to communicate cultural aspects by presenting two experiments in which the following factors will be discussed:

1) The experience prototyping activity associated to moments favourable to express emotions leads the individual to tell real stories that help designers create efficient interaction scenarios. On this work it is understood as efficient interaction scenarios the ones whose use situations are adequate to the users' experiences of the researched context and the objective target of the project's development. The moments of interaction were in groups when using the iTV applications followed of those of discussions on focus group; and

2) The theatre technique helps HCI professionals on the development of an effective communication process over the purposes, interaction possibilities, benefits provided and on the identification of the beneficiaries themselves through the interaction scenarios. By effective communication it is understood a process in which occurs the supply or exchange of information, ideas and feelings, through written or spoken words, or through signals, resulting on a reciprocal comprehension and shared sense.

The activities that will be described next are part of a process of definition and communication of the users' needs for products that will be generated by a research project [18]. It is a multi-cultural project with stakeholders from 9 European and Brazilian organizations. This project aims at creating a computational environment that allows citizens to have access to contents produced by the population through the WEB. In the Brazilian context, these citizens are users residing in the city of Barreirinhas, a small municipality (47.728 inhabitants in the state of Maranhão) [14]. This city is destitute of spaces that allow direct and continual contact with computers, and it's socio-economic condition, in a general way, does not favour the acquisitions of more sophisticated technological devices. The created content will be available through services that will be accessed by iDTV and/or mobile devices. Therefore, the stakeholders had to discover which services would be more interesting to the communities. Next we will present two experiments: the first one concerns the participation of the users and takes place in their real context of use, and the second one concerns the communication among all the stakeholders at the laboratory site.

4 Experiment 1

4.1 Method and Participants

The experiment was done from January 26th to 31st 2007, by the following team of experts: a psychologist, a designer and a user experience expert. It occurred in two stages, after the recruiting and the contextual analysis were applied: 1) experience prototyping with TV and Computer by applying test scenarios and 2) gathering of community' needs and expectations. In the first stage, the team simulated situations in which participants could perceive, feel and know several iDTV use possibilities. The objective of the team in doing use situations simulation was to activate the sensibility of the participants, so that they could, from their lived experience, apprehend and elaborate cognitions about iDTV, being able to reflect and criticize it, bringing this process to their reality.

By the occasion of the gathering of the community's needs and expectations regarding iDTV, focal groups were used. In order to facilitate the exposition of subjects' speeches, it was applied the technique of photolanguage, using drawings of faces that represented several expressions: sadness, happiness, surprise, among other (see Figure 1).



Fig. 1. Drawings of faces

The study had a total of 22 (twenty two) participants who live in the mentioned city. Their profile has shown to be very diverse, since it was formed by individuals from 8 (eight) years old to 55 (fifty five) years old, with educational level varying from elementary school to post graduation, the income of the sample group is situated between less than one minimum wage to above 10 minimum wages, and the diverse professional categories include: 05 students from elementary and junior high school, 06 teachers, 04 public employees, 03 salesmen, 03 people connected to tourism, and 01 community leader.

4.2 Scenarios and Stories in Experience Prototyping

On experience prototyping, the individuals were asked to perform one or more tasks. By choosing a scenario to vote in hypothetical programs, and using the t-vote application for iDTV, a possible scenario could be: Today is the last day to vote to eliminate a candidate from Big Brother. Access the electronic voting application available on your TV to eliminate the candidate that least pleases you on the House. Figure 2 illustrates an individual doing a scenario [20]. During the sessions, each with an average of 30 minutes, the participants were always accompanied by a member of the research team for any clarification.



Fig. 2. Scenarios of test being performed on DTV

Participants had the opportunity to speak and express their feelings. The analysis process of users' emotional responses (comments and stories) collected during the stimulus (the simulated situations and focus group) helped the team to interpreter their needs, as well as the city's. The needs were considered representative of the city because participants expressed their expectations regarding iDTV, always taking into account the city´s difficulties. Stories and comments were organized in topics following the criteria of highest frequency present on the individuals' speeches. The topics of needs were: the need of interaction; the need of communication; the possibility of elaborating and having access to information; and the need of entertainment with interactivity (as voting, games). We identified a set of possible iDTV services to attend the multi-cultural needs of users and defined the users' profiles (personas). The personas represent categories of potential iDTV service users.

In this experiment, the emotional responses could be associated to three aspects: the nature of the iDTV applications (such as modernity and innovation); the use of the applications (such as difficulties to use the remote control); and the meaning of the applications iDTV (such as the benefits of technological convergence) (see Figure 3).

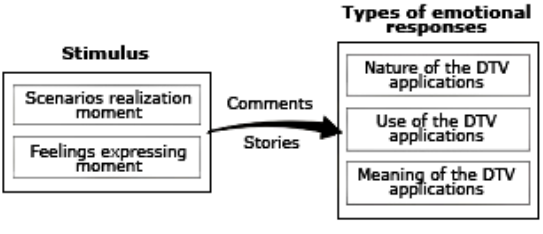


Fig. 3. Stories and comments to express emotions

For the need of elaborating and having access to information an example of a comment associated to the first aspect can be: "I'm so happy because I can keep up to date with what is happening". An example of a comment associated to the use of the application can be: "Writing a message on the web to be visualized on TV, will make it easier the communication with friends that don't have a computer". In relation to the third aspect, a participant (a teacher) told a story as a way of communicating how happy she was to know that a content created by her could be seen on TV by her students. This functionality would be particularly important to students that live far away and that miss class. The teacher would write the class on a notebook and ask a

colleague that lives next to the student that missed the class to take him the notes and homework; otherwise, the student wouldn't have access to the content given on the missed class. Therefore, such resource would implicate on a new attitude from the teachers and the relation family-school.

4.3 Scenarios in Interaction Design

Scenarios of interaction were constructed to represent the interaction of iDTV services, through the realization of the following steps:

1. Definition of the elements of an Interaction Scenario (IS), such as: scenario objective, involved personas, actions of the personas using the technology in several environments and, finally the types of contents manipulated; and
2. Association of these elements to the artifacts resulted from the users' study, such as the stories, the personas, etc.

The elaborated scenarios contain a narrative from 4 to 7 lines long. Each storyboard represents a scenario that contains an average of four frames. Different from the stories told by the users on the first experiment, which were many times not structures, the interaction scenarios narratives present a beginning, a middle and an end – that is, a complete text structure.

A document of 65 pages, a deliverable for the project, was elaborated describing the entire process with the results achieved. On the following 15 days from the document review, we would have a meeting in Brazil with all the partners (n= 19). On this event, the results of the users' study would be presented. Instead of focusing on the verification of the interaction scenarios, that been already been made, we wanted to effectively communicate interactive design solutions that brought value to the real needs discovered. For such communication, the following challenges were distinguished: 1) there was a problem of understanding the following terms: TV applications, TV services, and scenarios; 2) many of them were not yet familiar with the reality of the studied users. It was necessary then to use a resource that would promote the homogenization of the language for the understanding of the purposes of the verified interaction scenarios.

5 Experiment 2

On the meeting, 27 (twenty-seven) stakeholders were present, of which 8 worked on the organization of this event and 4 were responsible for the definition of the interaction design of this project. On the first day, after a brief presentation of the results of the study on the researched communities, we invited the participants to join us at the theatre to know the users' profiles and the intention of use of project's products. No additional information was given to the participants about them or about iDTV services to be offered. The theatre presentation was 30 minutes long. When the presentation was over, there was a moment of great emotion from the surprise of the artistic production. On the occasion the actors involved on the presentation were acknowledge. Next, all the participants returned to the meeting room and a discussion was initiated and a questionnaire was applied. At the end of this discussion session, a questionnaire comprised of 5 open-ended questions was applied. The questions were

related to the importance of the strategy applied to the understanding of aspects of the interaction scenarios, and the services to be offered. Only 15 participants answered the questionnaire, since the other 12 were directly involved in the interaction design and communication strategy.

The whole description of this experiment is described in [23], specifically, how the stories were (re)constructed. One step performed was the definition of the elements for the production of the theatre presentation (such as the script, which relates the story to be interpreted, added to interpretation resources, such as demonstration, error treatment support, help). It is important to point out the stories to be interpreted had a narrative of approximately 15 lines. Different from the narratives of the interactive scenarios, on the theatre stories there was a description of how the services helped the involved not only as a inclusion mean, allowing or facilitating the use of technology, but mainly as an informative source of how iDTV interactivity could aid their activities adding value to their everyday life. Therefore, a story (the script), contained interactive details that promoted cultural changes, inspired on the involved personas.

6 Results

In order to assess the impact of the stories on the interaction design process, we observed the effects of the real stories told by final users for the creation of efficient interaction scenarios (Factor 1), as well as the effects of the telling stories on the theatre in order to improve the understanding of interaction scenarios purposes by all the involved on the project (Factor 2).

6.1 The Impact of Real Stories on the Creation of Interaction Scenarios

Initially, we need to show in this analysis that the stories were representative of the researched context. Within the 22 participants of the first experiment, 7 participants told stories. Although the 7 stories represented only 31% of the participants, the stories were usually emphasized by the other participants through their comments. The 34 comments registered from the other participants (n=15) all related to the stories. Within the comments, 29 % (n=10) were negatives (e.g. It is stressing to understand the keys in English of remote control) and were all related to problems present on the stories. 71 % (n=24) of the positive comments were about their feelings and expectations towards iDTV (e.g. “Now it will make easier the communication with friends”) and were related to needs present on the stories. The social and economic problems within the stories were also pointed out on the report of the development of the city, studied and produced by IBGE [14].

In addition we also need to show the utility of the stories to the delimitation of the project’s scope. That is why a story is considered useful to generate efficient scenarios if it contemplates iDTV services that are according to the target objectives of this project. All the stories were useful to the identification of services that attend the mentioned needs and that are within project’s scope. For the enunciated stories, 5 services were identified: 1) discussion, which allows the interaction among viewers through the TV; 2) visualization of information on TV; 3) surveys, to promote the

participation of the community, 4) entertainment – manipulating digital pictures, to guarantee involvement with the digital content, and 5) electronic commerce. Figure 4 illustrates the relation among real stories and services. Each story highlighted at least one service, pointing out that 3 stories staged 3 services and 3 stories highlighted 2 services. We call attention to the fact that the information visualization service (service 4) is pointed out in 5 of the 7 stories told. Such result was considered on project's findings.

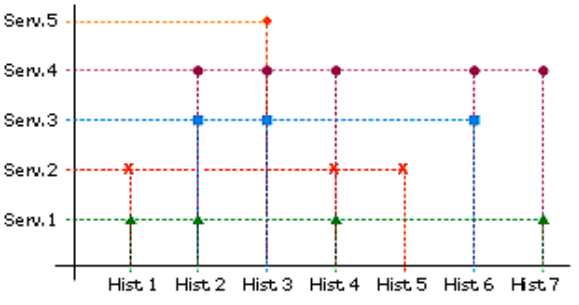


Fig. 4. Relation among real stories and services

Service 5 (electronic commerce) mentioned on story 3 was discarded, since it wasn't within project´ scope. Figure 4 shows that the scope of the story 3 involved other services, and therefore the discarded service did not implicated on the disregard of the entire story.

Considering that the stories were representative of the context, all of them guided Brazilian designers in some way (including the one that participated on the first experiment) to create efficient interaction scenarios. 6 scenarios were inspired in all 7 stories. The story narratives were primordial to the definition of scenarios' objectives and to the identification of the beneficiaries/involved (institutions and citizens). All 6 personas were considered on the definition of the theatre's characters and participated in at least one scenario.

6.2 The Impact of the Theatre Technique as a Resource to Communicate Objectively the Purposes of Interaction Scenarios

The speeches provided by the stakeholders (which were collected by the application of a questionnaire after the theatre presentation) were analyzed. This analysis was performed by taking into account the following categories: understanding of real; ways of communication, involvement with the content; particular interaction aspects; and clarification of each term. In a simplified way we can highlight the following issues.

From the positive comments of stakeholders it was possible to realize their understanding concerning the services to be offered with the future products. Other participants felt as if the hypothetical situations presented in the theatre performance were real, thereby establishing diverse connections with the concrete situations. The phrases: "It shows real life" and "The scenarios were quite realistic" said by two of stakeholders can confirm the foregoing definition.

In the category *Communication Type,* several aspects were considered, such as objectivity and clarity of communication, in order to avoid blockage, noise and filtering-elements that characterize a type of communication that does not occur in a proper way or that is not successful in its objective of communicating. It is understood that the rhythm in which the scenarios were presented contributed to the pragmatism, objectivity, and quickness of the message. We were able to observe the valorization of these requisites by two speeches´ stakeholders: "It is more direct, easy, engaging with the content than a written explanation."

Referring to the category *Involvement with the Content*, this is observed in the audience's feeling of credibility in the level of development of this study, their opinion about the characteristics of Project and their feeling of valorization for being part of an original study. Therefore, we point out the following comment: "I felt comfortable seeing images in action a plus words (and not just words), because it made the transmission of ideas and concepts clear, obtained during the process of defining users' needs (the results of users' field studies)".

7 Discussion and Future Works

7.1 Technological Insights from Stories

Since the test scenarios were used as a support technique to the exploration of new ideas, and the answers of the users were expressed on the stories, it is left to us to verify if on the speeches there were useful insights into this novel domain. To some of the stories told, the technological suggestions provided the storytellers themselves:

- The rural teacher suggested that the grades and/or teaching material, which the school sends by mail, could be visualized via iTVD, but only to those that show interest. This was a solicitation of a requisite of the treatment of users' preferences (story 1);
- The storyteller of story 2 suggested that when the user sent a message via iTVD and it were off, that it would have a way of informing that it was off to sender of the message, and a new message alert to the person that the message was sent to and;
- The storyteller of story 7 suggested that there was a support to the many people that didn't know how to type the text, via Remote Control (RC). The difficulty of handling the RC was actually acted out by one of the actors during the theatre presentation. On this storyteller case, he suggested voice command activation.

The suggestions were motivating to the team, which realized the importance of applying user experience and storytelling techniques. The simple application of the traditional techniques of data collecting (as questionnaires) wouldn't have led the participants to relate real situations with imaginary solutions of iDTV services.

7.2 Storyboards and Theatre as Complementary Strategies

From comments given by the participants after the theatre realization, it was fairly easy to realize this last technique must be applied as a complementary strategy to

storyboards. The comments focused on understanding of users' preferences, their limitations and their emotions instead in technical comments. We could suppose that it can be because, for instance, on the interaction scenarios storyboards, it is represented the convergence of the illustrated medias, the information needed to interact, which give space to technical comments. On the theatre stories, however, it is shown that the convergence will not be only technological. Through interpretation resources, it was shown that there will be future spaces of interaction for a cultural convergence. This means that there will be several ways of creating and disseminating knowledge, besides providing collaborative processes of content production and TV democratization. Such supposition was partly confirmed by S 2, mentioned previously.

Three months later, a video was produced by illustrating each scenario represented in the performance with storyboards. The video was showed to users in other experiment [10].

7.3 Vote of Trust

There is a situation that demonstrates that there was credibility of the stakeholders for the work presented here. This situation was perceived taking into account the suggestions given after the elaborated document. Some suggestions were related to the inclusion of new iTV services, such as: pizza delivery, product order and security control of the city. On that time we answered that such suggestions would be discussed on the meeting. By the occasion of the meeting, two participants answered the questionnaires saying that they weren't certain that if all scenarios would fit to the user's needs, and suggested to validate the work with the final users themselves. Apart from this, there were no suggestions for the inclusion of new services.

As weakness of this approach we highlight that we should have invited the users to participate on the meeting, representing at least one of the personas. We admit that the inexperience with the application of the theatre technique and the successful use of storyboards on previous projects did not make us reflect about the advantages that the meeting of stakeholders and users could bring to the validation of the services. Other reason was that we were more worried about the difficult communication among stakeholders, and so, we had to first provide a common understanding.

Therefore we promoted eight months latter a meeting in the mentioned city with the stakeholders (n=10). We are still analyzing the impacts of the theatre representation in relation to what they perceived in city, such as the identification of the users that were in the meeting with the personas; and the relation of the developed applications with the represented and validated scenarios.

8 Conclusions

In this work we studied how the relation between scenarios and stories would have an impact on exploration and communication of interaction design, by developing three different types of relations:

- Test scenarios were done by users stimulating the appearance of real stories. By the occasion of the scenario realization, we created opportunities closer to the everyday life of the researched individuals, by involving them and stimulating

them in imagining the several iDTV possibilities. The participants had insights and told real stories when associating the situations lived by them on their everyday life to the simulated situations;

- Real stories were useful artifacts for the elaboration of interaction scenarios. The stories and also the comments of the participants describing expectations and emotions regarding the technology allowed HCI professionals to learn their personal experiences and to feel what they really expected of the interactive services, projecting efficient interaction scenarios; and
- Interaction Scenarios were transmitted to the involved through stories told by theatre technique. On this relation, it was proved that the stories told by theatre technique favoured the understanding of all about the objectives and beneficiaries of the system. On the theatrical representations were demonstrated HCI purposes for social and digital inclusion. The quality of the theatrical play led stakeholders to believe on the potential of this interaction project.

Acknowledgements. This work was funded by the IST EC-project SAMBA.

References

1. Beers, R., Whitney, P.: From Ethnographic Insight to User-centered Design Tools. In: EPIC 2006, pp. 139–149 (2006)
2. Bettelheim, B.: A Psicanálise dos contos de fadas. R.J.: Editora Paz e Terra (1979)
3. Buchenau, M., Suri, J.F.: Experience Prototyping. In: Proceedings of the conference on Designing interactive systems: processes, practices, methods, and techniques, New York City, New York, United States, August 17-19, 2000, pp. 424–433 (2000)
4. Busatto, C.: A arte de contar histórias no século XXI: tradição e ciberespaço. Vozes, Petrópolis (2006)
5. Carroll, J.: Making use: Scenario-based design of human-computer interactions. MIT Press, Cambridge (2000)
6. Carvalho, A., Mendes, M., Pinheiro, P., Furtado, E.: Analysis of the Interaction Design for Mobile TV Applications based on Multi-Criteria. In: CONFENIS 2007 (2007)
7. Cooper, R.: The Essentials of Interaction Design. Wiley, Chichester (2003)
8. Dow, S., Saponas, T., Li, Y., Landay, J.: External Representations in Ubiquitous Computing Design and the Implications of Design Tools. In: DIS 2006, pp. 241–250 (2006)
9. Eronen, L.: User Centered Design of New and Novel Products: case digital television. Helsinki. PhD Thesis – Helsinki University of Tecnology (2004)
10. Furtado, E., Schilling, A., Fava, F., Camargo, L.: Promoting Communication and Participation Through Enactments of Interaction Design Solutions - A study case for validating requirements for Digital TV. In: International Conference on Enterprise Information Systems - ICEIS, Barcelona (2008)
11. Gilmore, D.: Understanding and Overcoming Resistance to Eghnographic Design Research. Interactions IX, 29–35 (2002)
12. Hachos, J., Redish, J.: User and Task analysis Techniques. John Willey and Sons, New York (1998)
13. Hutchinson, H., Bederson, B., Druin, A., Plaisant, C., Mackay, W., Evans, H., Hansen, H., Conversy, S., Lafon, M., Roussel, N., Lacomme, L., Eiderback, B., Lindquist, S., Sundblad, Y., Westerlund, B.: Technology probes: Inspiring Design for and with Families. In: CHI 2003, pp. 17–24 (2003)

14. IBGE Instituto Brasileiro de Geografia e Estatística – IBGE. Censo Demográfico (2000), http://www.ibge.gov.br/home/estatistica/economia/perfilmunic/2002/default.shtm (accessed, 05/26/2007)
15. Mattos, L.: Uma metodologia para formação continuada de professores universitários no contexto de um sistema multiagentes. Máster Dissertation. UFC (2001)
16. McCarthy, J., Wright, P.: Technology as experience. In: MIT 2004(2004)
17. Preece, J., Rogers, Y., Sharp, H.: Design de interação: além da interação homem-computador. Bookman, Porto Alegre (2005)
18. Project SAMBA. System for Advanced interactive digital television and Mobile services in BrAzil. Disponível em (2007), http://www.ist-samba.eu/
19. Pruitt, J., Adlin, T.: The Personal Lifecycle: Keeping People in Mind Through out Product Design. Elsevier, Amsterdam (2006)
20. Soares, P., Valente, D., Mendes, M., Furtado, E.: Uma Ferramenta para suportar a convergência da TV digital com a web a partir de uma análise de possíveis situações de uso. In: CLEI 2007(2007)
21. Sutcliffe, A.: User-Centred Requirements Engineering – Theory and Practice. Springer, Heidelberg (2003)
22. Truong, K., Hayes, G., Abowd, G.: Designing interactive systems. In: DIS 2006, pp. 12–21 (2006)
23. Users' Needs including Requirements Specification Document. Delivrable D 2.1., IST EU FP6 SAMBA Project (May 2007)

End-User Development for Individualized Information Management: Analysis of Problem Domains and Solution Approaches

Michael Spahn[1] and Volker Wulf [2]

[1] SAP AG, SAP Research, Bleichstr. 8, 64283 Darmstadt, Germany
Michael.Spahn@sap.com
[2] University of Siegen, Hölderlinstr. 3, 57076 Siegen, Germany
Volker.Wulf@uni-siegen.de

Abstract. Delivering the right information at the right time to the right persons is one of the most important requirements of today's business world. Nevertheless, enterprise systems do not always provide the information in a way suitable for the individual information needs and working practices of business users. Due to the complexity of enterprise systems, business users are not able to adapt these systems to their needs by themselves. The adoption of End-User Development (EUD) approaches, supporting end-users to create individual software artifacts for information access and retrieval, could enable better utilization of existing information and better support of the long tail of end-users' needs. In this paper, we assess possibilities for improving information management through EUD, by analyzing relevant problem domains and solution approaches considering fundamental aspects of technology acceptance theories. The analysis is based on a questionnaire survey, conducted in three midsized German companies. We investigate the domains of information access and the flexible post-processing of enterprise data. Therefore we assess the importance of the respective domain for the work of end-users, perceived pain points, the willingness to engage in related EUD activities and the perceived usefulness of concrete EUD approaches we developed to address the respective domains.

Keywords: Information Management, End-User Development, Information System Design, Empirical Studies.

1 Introduction

In our today's business world, companies use enterprise software systems like Enterprise Resource Planning (ERP) systems to support and facilitate their business processes. As companies are not static and evolve like the environment of competitors, markets and customers surrounding them, a continuous need of adapting these systems to new requirements, business processes and associated information needs, exists. Information management (IM) as a sub-area of business administration is concerned with making the best possible use of the resource information with regard to the company's objectives [1]. For this, IM has to plan, organize and control the balance of supply and demand of information, the information systems used to process

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 843–857, 2009.
© Springer-Verlag Berlin Heidelberg 2009

information, and the information and communication technology used as an infrastructural base. One major challenge of IM is to satisfy the information needs of virtually all stakeholders to enable effective and efficient processes within the organization, and to simultaneously cope with changing requirements.

Viewed from a micro-perspective, business users as the end-users of enterprise systems know best about needed adaptations and information [2], as they are managing and executing business processes on a daily basis. As end-users are domain experts but not necessarily IT professionals, they are not able to adapt the used enterprise systems to their individual information needs and working practices on their own [3, 4]. End-users are forced to indirectly "influence" the adaptation processes by communicating their needs to IT professionals. This puts end-users solely in the role of requirement givers, but does not enable them to actively take part in the development process. IT professionals on the other hand are confronted with requirements, which are expressed in the users' domain language and have to be interpreted and transformed into models and technical solutions matching the capabilities of the enterprise software systems. This process is not only costly and time-consuming, but also error prone due to a possible misinterpretation of requirements [5, 6]. Furthermore, many beneficial adaptations addressing changed processes, information needs, missing functionality or individual working practices are not realized due to limited budget, resources and expertise. One approach to improve this situation is to enable end-users to better adapt the used enterprise systems on their own. At this, the inherent challenge is to reduce the expertise tension, existing in a two-dimensional continuum of job-related domain knowledge and system related development knowledge [7].

We are approaching this challenge from an End-User Development (EUD) perspective. EUD can be defined as "a set of methods, techniques, and tools that allow users of software systems, who are acting as non-professional software developers, at some point to create or modify a software artifact" [8]. The adoption of EUD approaches could enable IM to better cope with the long tail of end-user's individual information needs and working practices. Allowing end-users to create individual software artifacts for information retrieval, merge and analysis, could enable a better utilization of existing information and result in a better balance of information demand and supply.

In order to assess possibilities for improving IM through EUD, it is first necessary to consider different aspects that influence the uptake of EUD in organizations [9]. As a precondition of end-users being willing to invest effort in EUD activities, the problem area addressed needs to be perceived as relevant and important for their work tasks. The uptake of EUD activities is positively affected if end-users have a positive attitude towards EUD and perceive EUD activities to be beneficial. According to the Technology Acceptance Model (TAM) of Davis [10], the perceived usefulness and ease of use of concrete EUD approaches also has to be considered, as they are important factors determining their acceptance.

In this paper, we verify and further analyze problem areas relevant for IM, which have been identified in qualitative preliminary studies [3]. We use quantitative methods and the instrument of a questionnaire to analyze the problem areas on a broader end-user base. We assess basic attitudes of end-users towards EUD activities, the

perceived difficulties in the respective problem areas, and perceived usefulness of approaches addressing the identified problem areas. The remainder of the paper is organized as follows. In Section 2 we present the theoretical background, basic findings of preliminary studies, and the setup of the conducted questionnaire. In Section 3 we present and discuss the results of the conducted questionnaire study. Finally we summarize and conclude the results in Section 4.

2 Background and Setup

In this section we discuss the theoretical background of the aspects we analyze, briefly discuss findings from our preliminary studies, and discuss the setup of the used questionnaire.

2.1 Theories of Technology Acceptance

For assessing the applicability of EUD approaches several fundamental technology acceptance aspects have to be considered. The Theory of Reasoned Action [11] postulates that voluntary behavior of individuals can be predicted through their attitude towards the behavior and the individuals' perception of how other people would view them if they perform this behavior (subjective norm). This theory is extended through the Theory of Planned Behavior [12], which introduces the concept of self-efficacy in addition to attitudes and subjective norms. Self-efficacy relates to the perceived behavioral control, i.e. it expresses the conviction of individuals that they can successfully execute the anticipated behavior. The concept of self-efficacy is also considered in the Social Cognitive Theory [13], where it is combined with the expectations of a valued outcome from a performed behavior. The provided psychological foundations are used to develop information systems theories for the acceptance and use of information technology. Such theory is the Technology Acceptance Model [10], which is based on the Theory of Reasoned Action [11] and replaces its attitude measures with two technology acceptance measures - perceived ease of use, and usefulness.

As intention is a key determinant of action [10, 11], and increased intention also implies increased uptake of EUD activities and the acceptance of according approaches, we assess aspects influencing the intention of end-users to engage in EUD activities in order to predict suitability of EUD approaches to support problem areas relevant for IM.

2.2 Preliminary Studies

In preparation of developing EUD approaches for IM, we conducted a series of 14 semi-structured interviews with ERP users, based on qualitative research methods [14] in three German midsized companies that use ERP systems to support their business processes. The addressed companies were two companies from production industry (137 and 140 employees) and one larger software vendor (500 employees). We identified two main problem areas that end-users face with regard to data-centric EUD activities. First, end-users need to access the same set of data using the graphical user interface (GUI) of the ERP system over and over again in a cumbersome manner,

and are not able to create custom, interactive applications that provide the information relevant for individual working tasks at a glance. Second, end-users create custom spreadsheets and rely on getting relevant business data from the ERP system by using queries, but face considerable challenges when trying to create custom queries for their individual information needs. For a detailed discussion of the findings we refer to Spahn et al. [3].

2.3 Setup of Questionnaire

To refine and verify the results of our preliminary studies on a broader end-user base, and simultaneously get feedback on the perceived usefulness of possible solution approaches, we conducted a questionnaire-based survey among the users of ERP systems in the companies that already participated in our preliminary studies. We addressed users of ERP systems, as ERP systems are central systems for IM and store the most important enterprise data, being relevant for a broad variety of employees and their individual working tasks. The questionnaire was provided online and consisted of three main parts. The first part focused on the users themselves, the software they use and their basic attitude towards adapting software systems to their individual needs on their own. The second part focused the problem area of accessing information in enterprise systems, while the third part focused the problem area of individually post-processing data from enterprise systems. The second part included a video presenting the prototype of an EUD environment for mashing up enterprise information to individual widgets for information access. Embedding the video allowed a first evaluation of the prototype with respect to perceived usefulness according to TAM [10].

To conduct the survey, a link to the questionnaire was sent via e-mail to one contact person of each participating company. The contact persons distributed the link to the questionnaire to randomly chosen users of ERP systems within the respective company. As the only criterion for participating in the survey was being an ERP user and the users have been chosen randomly, the results reflect the options and attitudes of a broad variety of ERP users of all kinds. We received 73 filled and analyzable questionnaires. With regard to a total of 236 ERP users in the companies, we achieved a coverage of 30.1% of the addressed test population.

3 Results

In the following sub-sections we present and discuss the results of the individual parts of the questionnaire.

3.1 Users and Software

The participants represent a well-balanced mix of gender, age and educational level (cf. Table 1). The diversity of participants indicates the consideration of the opinions of a broad variety of ERP users. Combined with the high coverage of 30.1% of the addressed test population of ERP users within the three participating companies, this indicates a good approximation of representative results.

Table 1. Gender, age and educational level of participants

Gender	male	62.3%
	female	36.2%
	No answer	1.5%
Age	<25 yrs	7.1%
	25-29 yrs	14.3%
	30-39 yrs	27.1%
	40-49 yrs	31.4%
	50-59 yrs	15.7%
	>60 yrs	0%
	No answer	4.4%
Level of education	Secondary general school	6.7%
	Intermediate school	10.0%
	Grammar school	18.3%
	Dual vocational training	31.7%
	University of applied sciences	13.3%
	University	16.7%
	PhD	3.3%
	No answer	0%

Table 2. Intensity of using ERP systems and office tools

How often do you use...	**never**				**very often**
	0	**1**	**2**	**3**	**4**
SAP ERP	21.9%	9.6%	13.7%	11.0%	38.4%
MS Navision	35.6%	16.4%	8.2%	5.5%	15.1%
MS Excel	1.4%	15.1%	26.0%	26.0%	30.1%
MS Access	43.8%	26.0%	5.5%	5.5%	5.5%
MS Word	1.4%	8.2%	26.0%	38.4%	23.3%
MS Outlook	0.0%	0.0%	4.1%	20.5%	74.0%

In our preliminary studies we observed, that many users just use the ERP system and standard office applications to accomplish their work tasks. To get more details on the usage behavior, we asked participants how often they use ERP systems and office applications (cf. Table 2). The intensity of usage can be seen as an indicator for the importance and relevance of the systems for the end-users with regard to their work tasks, and accordingly the importance and relevance of considering these systems and processed data within EUD approaches. We asked participants for additional tools they use for their work tasks, but received only few answers, indicating a confirmation of the assumption that participants mainly use ERP and office applications.

Table 2 shows the intensity of using ERP systems and office applications, as stated by the participants. The used scale was never (0), rarely (1), regularly (2), often (3), and very often (4). Please note, that in Table 2 and all following tables, we omitted the percentages of users who declared not to answer a questions for the sake of compact overviews. Thus, missing percentages to 100% have to be interpreted as the percentage of users who declared not to give an answer. All participating companies use SAP ERP as an ERP system, and one participating company additionally uses

Table 3. Desirability and perceived usefulness of being able to adapt software systems

Would you like to be able to better adapt the used software systems to your individual needs on your own?	**No, not at all**				**yes, a lot**
	0	**1**	**2**	**3**	**4**
	4.1%	8.2%	38.3%	30.1%	15.1%
Do you think you could better execute, ease or speed up your work, if you were able to better adapt the used software systems to your individual needs?	**no benefit**				**high benefit**
	0	**1**	**2**	**3**	**4**
	5.5%	6.8%	34.3%	37.0%	13.7%

Microsoft Navision, which we included for the sake of completeness. As a consequence there is a certain percentage of participants that declared not to use SAP ERP or MS Navision, which is caused to a great extend due to the fact that these participants mainly use the other ERP system, and thus cannot be seen as an indicator for a certain percentage of participants not using ERP systems at all. 63.1% of participants use SAP ERP regularly, often or very often, and 28.8% use MS Navision at least regularly. With regard to the above mentioned bias, the overall intensity of ERP usage has to be considered to be high. To avoid system specific biases, we subsumed both ERP systems using the umbrella term enterprise system in the rest of the questionnaire.

In the questionnaire we used names of popular office applications as representative of the respective kind of application, but clearly stated that usage of other applications of the same kind should also be considered (e.g. if using a spreadsheet application other than Excel). 98.6% of participants stated to use Outlook at least regularly, 87.7% use Word, 82.1% use Excel, and 15.5% use Access at least regularly. Considering the usage intensity, Outlook, Word and Excel can be regarded as fundamental tools for the participants in the context of their work tasks. With regard to data-centric applications, Excel is much more used than Access, and nearly a third of the participants (30.1%) state that they use Excel very often, indicating its importance for data-centric tasks and the importance of flexibly processing data for the work of participants.

A crucial precondition of the acceptance of EUD approaches is the attitude of end-users to perceive the ability of being able to adapt software systems on their own desirable as well as useful. To assess the basic attitude of end-users, we asked the participants if they would like to be able to adapt software systems to their individual needs, and if they think that this would ease or speed up their work. 83.5% of the participants rated the desirability of adapting software systems on their own with at least 2 on a scale ranging from 0 (no, not at all) to 4 (yes, a lot). 85% of the participants rated the usefulness of being able to adapt software systems with at least 2 on a scale ranging from 0 (no benefit) to 4 (high benefit). This indicates that the majority of end-users perceive the ability of adapting software systems on their own desirable as well as useful.

With regard to the attitude of end-users, an interesting aspect is, if this attitude is based on personal experience with EUD activities and the perceived valued outcomes, and if EUD activities are already common. Using creation and adaptation of spreadsheets as an exemplary case of EUD activities, we asked the participants to which extent they create individual solutions for their tasks using spreadsheet applications, and to which extent they just consume spreadsheets created by others. 75.3% of the participants stated to create own solutions or modify existing solutions, whereas only 28.8% of

Table 4. Features of Excel used by participants

Which features of Excel do you use?	**never used**	**already used**	**regularly use**
Enter data	1.4%	19.2%	76.7%
Formatting data (changing visual appearance)	2.7%	20.5%	71.2%
Formulas like ”=A1+B1”	4.1%	28.8%	63.0%
Formulas using functions like ”=sum(...)”	4.1%	26.0%	65.8%
Formulas using control structures like “=iif(...;...;...)”	24.7%	43.8%	20.5%
Pivot tables	42.5%	31.5%	12.3%
Formulas using VLOOKUP / HLOOKUP function	47.9%	27.4%	8.2%
Recording macros	60.3%	21.9%	2.7%
Create or modify macro source code	61.6%	20.5%	2.7%

participants stated to consume existing spreadsheets of others. This indicates that EUD activities are quite common. As an indicator for the ability of end-users to create even complex solutions, we assessed if only basic features are used in EUD activities, or if end-users also use advanced or programming-like features. We asked the participants which features of Excel they use (cf. Table 4). All listed features, even the advanced features, have already been used by at least a fifth of participants, whereas the more basic features are more regularly used in the everyday work. This indicates that end-users are able to create even complex solutions using EUD activities.

In summary the results show that end-users use mainly ERP systems and office applications to accomplish their working tasks. It is common for end-users to create own solutions for their individual needs, as exemplarily seen in the creation of individual spreadsheets. They are willing to perform EUD activities and perceive such activities as being useful and beneficial. The observed attitude and practices can be seen as a beneficial base for adopting new EUD methods and tools. In the following we investigate two problem areas end-users face in more detail and present EUD approaches addressing these problem areas.

3.2 Accessing Enterprise Data

In our preliminary studies we observed that end-users are usually not able to adapt the GUI of the used enterprise systems. With regard to accessing live data of enterprise systems, end-users are forced to use the predefined structure of the GUI, which does not always provide the data in a way suitable for the individual tasks and information needs of end-users. We included questions in the questionnaire to assess the importance of information access, practices of information access, and if participants perceive information access to be cumbersome and wish to ease the access. According results are presented in Table 5.

82.2% of participants state that their work regularly, often or very often depends on accessing data within enterprise systems, indicating a high importance of information access for the executed work tasks. With regard to the access behavior that occurs regularly, often or very often, 80.7% of the participants declared that they need to retrieve the same information in the same way. 67.1% need to gather information step by step. 56.1% think that accessing the needed information is cumbersome, and

Table 5. Results of questions regarding access of information within enterprise systems

How often...	**Never**				**very often**
	0	**1**	**2**	**3**	**4**
…your work depends on accessing data within enterprise systems?	1.4%	16.4%	15.1%	23.3%	43.8%
...do you retrieve the same kind of information in the same way?	0.0%	15.1%	20.5%	30.1%	30.1%
...do you need to gather information step by step?	1.4%	28.7%	32.9%	27.4%	6.8%
...do you think that accessing the information you need is cumbersome?	1.4%	39.7%	34.2%	20.5%	1.4%
...do you wish that accessing needed information would be easier?	0.0%	30.1%	34.2%	23.3%	9.6%

Table 6. Perceived usefulness of individually tailored applications and willingness to accept learning efforts to enable their creation

Do you think you could ease or speed up your work, if an application tailored to your needs presents all needed data at a glance?	**no benefit**				**high benefit**
	0	**1**	**2**	**3**	**4**
	2.7%	8.2%	32.9%	38.4%	16.4%
Would you be willing to accept learning efforts to be able to create applications like in the last question by yourself?	**no**	**0.5 h**	**>1 h**	**1 d**	**>1 d**
	6.8%	5.5%	26.1%	19.2%	35.6%

67.1% wish that accessing needed information would be easier. The results reveal that the work of end-users depends on accessing data within enterprise systems, and that end-users have to retrieve the same information recurrently, often in a stepwise manner, which is perceived to be cumbersome and results in the wish for easier access to information.

One way of supporting end-users is to provide EUD tools enabling the creation of custom, interactive applications that provide the information relevant for individual working tasks at a glance. We asked participants about the perceived usefulness of such applications, and if they were willing to accept learning efforts to create such applications (cf. Table 6). Asked about applications tailored to their needs, that present all needed data at a glance, 87.7% of the participants rated their potential benefit to ease or speed up their work to be at least 2 on a scale ranging from 0 (no benefit) to 4 (high benefit). 80.9% of participants would accept learning efforts of several hours up to several days to be able to create such applications. This indicates a high perceived usefulness and willingness to invest effort in the creation of such solutions.

3.3 EUD for Information Access

Based on the insights of our preliminary studies, we set up a prototypic EUD design environment called “Widget Composition Platform” (WCP), based on an early internal prototype of SAP Research, that we branched and extended to suit our needs. The

WCP is a web-based EUD environment that enables business users to mash up enterprise resources in a visual design environment in a very lightweight way and to deploy the created mashups in form of widgets to their local machines. By using a very lightweight, visual WYSIWYG mashup design paradigm and encapsulating mashups as widgets, business users are enabled to develop small, interactive applications from enterprise resources and to deploy these applications directly to their desktop, without the need of any programming knowledge. As an example the WCP enables the users to drag & drop enterprise data like customer master data and sales orders to a design space representing the widget. By connecting the customer data with the sales order data using a single line, the sales orders are restricted to the sales orders of the currently selected customer. Using this paradigm, end-users are able to create custom interfaces to data that might be widely distributed within the ERP systems within minutes. For the sake of brevity we refer to Spahn and Wulf [15] for a detailed description of the WCP.

The questionnaire included a video presenting the WCP. The video showed the creation of a widget according to a previously presented use case and carefully considered showing all details of the creation process to enable participants to reasonably rate the complexity of widget creation. After showing the video, the questionnaire asked several questions directly related to the WCP approach (cf. Table 7). According to Davis [16] video presentation can be seen as a viable medium for demonstrating systems in a user acceptance testing context as it enables subjects "to form accurate attitudes, usefulness perceptions, quality perceptions and behavioral expectations (self-predictions of use)".

On a scale from 0 (too difficult) to 4 (very easy) 82.8% of the participants rate the difficulty level of widget creation using WCP to be 2 (passable), 3 (easy) or 4 (very easy). 72.5% think that they can manage to create widgets on their own using the WCP environment, thus expressing a high degree of self-efficacy. 69.1% think that custom-made widgets are able to provide benefits in real work context, indicating a positive expectation of a valued outcome. This indication was amplified by results indicating a high perceived usefulness, e.g. on a scale ranging from 0 (widgets do not provide any benefit for me) to 5 (widgets would help me a lot), 60.6% of participants rated the ability of widgets to ease or accelerate their own work with at least 3. 47.9% of the participants state, that a typical, personal work situation spontaneously comes to their mind, in which widgets would be helpful. 58.5% of these participants would be willing to create a widget on their own for that work situation, thus expressing increased intent to perform EUD activities using the WCP. This intention is amplified by results showing that 88.3% of the participants are willing to accept learning efforts to learn the creation of widgets, whereas 69.2% of these participants would accept learning efforts ranging from several hours to several days. 84% of the participants would use complete, predefined widgets in their everyday work, if they provide data relevant to them. In summary, the results suggest a high degree of perceived usefulness, perceived ease of use as well as positive attitudes towards using WCP, which are the major factors affecting technology acceptance according to the TAM [10].

Table 7. Questions addressing WCP and according results

Question	Results
How do you rate the difficulty level of the shown method of widget creation?	0% too difficult, 13.8% difficult but manageable, 39.7% passable, 22.4% easy, 20.7% very easy
Do you think you could manage to create a widget using the shown method on your own?	72.5% yes, 6.9% no, 17.2% uncertain
Do you think custom-made widgets can provide benefits in real work contexts?	69.1% yes, 5.3% no, 21.3% uncertain
Do you think, you could ease or accelerate your work by using widgets tailored to your needs?	Scale from 0 (would not provide any benefits for me) to 5 (would help me a lot): 6.4% 0, 5.3% 1, 17.0% 2, 26.6% 3, 26.6% 4, 7.4% 5
Is there any personal, typical work situation that spontaneously comes to your mind, in which a widget would be of help for you?	47.9% yes, 27.6% no, 14.9% uncertain
If your last answer was "yes", would you be willing to create a widget for that purpose on your own?	58.5% yes, 8.5% no, 17.0% uncertain
Would you be willing to accept learning efforts to learn the creation of widgets?	5.3% no, 7.4% yes, up to half an hour, 11.7% yes, up to an hour, 19.1% yes, several hours, 21.3% yes, a day, 28.8% yes, several days
Can you think of using complete, predefined widgets in your everyday work, if they provide data relevant to you?	84.0% yes, 2.1% no, 8.5% uncertain

3.4 Post-Processing Enterprise Data

In our preliminary studies, we observed that end-users create custom spreadsheets to flexibly post-process enterprise data for their individual needs. At this, they rely on getting relevant business data from the ERP system using queries, but face considerable challenges when trying to create custom queries for their individual information needs. We investigated the importance of flexibly post-processing data using spreadsheets for the work of end-users, and if they wish to be able to include more or different data from enterprise systems (cf. Table 8). Additionally we investigated the role of queries in this context, perceived usage problems and the desirability of easing query creation.

91.8% of the participants rate the importance of Excel for their job to be at least 2 on a scale from 0 (absolutely unimportant) to 4 (absolutely necessary), indicating a high relevance of Excel and the need to post-process data. The majority (55.6%) of participants spends 1% to 20% of work time with post-processing data from enterprise systems within Excel, and 30.6% of participants even spend 21% to 40% of their work time, which is a significant amount of time. 63.9% of participants use Excel to post-process data from enterprise systems regularly, often or very often. While 58.3% of participants are able to import data from enterprise systems regularly, often or very often, 20.8% of participants manually transfer data from enterprise systems by typing

Table 8. Importance of flexibly post-processing enterprise data using custom spreadsheets

Question	Results
How important is Excel for your job?	Scale from 0 (absolutely unimportant) to 4 (absolutely necessary): 1.4% 0, 5.5% 1, 21.9% 2, 31.5% 3, 38.4% 4
What percentage of your work time do you spend with post processing data using Excel?	5.6% (none or nearly none), 55.6% (1% - 20%), 30.6% (21% to 40%), 4.2% (41% to 60%), 2.8% (61% to80%), 1.4% (81% - 100%)
How often do you process data from enterprise systems using Excel?	13.9% (never), 16.7% (rarely), 23.6% (regularly), 18.1% (often), 22.2% (very often)
How often do you transfer data from enterprise systems in the following ways?	Condensed results of answers in range of regularly, often and very often: 20.8% typing data from screen or printout, 58.3% importing data from enterprise system, 41.7% receiving data as a file from colleagues.
Would you like to be able to transfer more or different data from enterprise systems to Excel?	Scale from 0 (no benefit for me) to 5 (would help me a lot): 16.7% (0), 13.9% (1), 8.3% (2), 33.3% (3), 12.5% (4), 6.9% (5)
Do you think you could ease or speed up your work, if you were better enabled to transfer the data relevant for you from enterprise systems to Excel?	Scale from 0 (no benefit for me) to 5 (would help me a lot): 15.3% (0), 9.7% (1), 8.3% (2), 25.0% (3), 20,8% (4), 11,1% (5)

data from the screen or printouts. With regard to the significant amount of working time many participants spend with processing data in Excel, this indicates a possible area of improvement. Asked if they would like to be able to transfer more or different data from enterprise systems to Excel, 52.7% rate the potential usefulness to be at least 3 on a scale from 0 (no benefit for me) to 5 (would help me a lot). Asked if they think they could ease or speed up their work, if being better enabled to transfer desired data from enterprise systems to Excel, 56.9% rate the potential usefulness to be at least three (same scale as last question).

In summary, Excel can be considered to be a very important tool for the participants, used in a significant amount of work time to post-process data from enterprise systems. Transferring data from enterprise systems is not always seamless, and the majority of participants would like to be able to transfer more or different data from enterprise systems to Excel, thereby easing or speeding up their work.

A standard way of transferring data from enterprise systems to Excel is to create queries within the enterprise systems and exporting the results to Excel. We included questions in the questionnaire to investigate how common the usage of queries is, and if its usage is considered to be problematic. For the sake of brevity we just present a short overview of related results in the following. 27.8% of the participants stated that they use queries to filter and display relevant data, 25% use queries for exporting data to Excel, 31.9% do not use queries, but know what queries are, and 18.1% do not know what queries are. We asked the participants using queries, how difficult they rate the creation or modification of queries on a scale ranging from 0 (very easy) to 5 (very difficult), and 32.2% of participants rate the difficulty to be at least 3. Asked, what aspects of query creation or modification are the most problematic, 50% stated that they

have problems to find data within the system (e.g. relevant tables or columns storing the data), 26.5% stated that they have problems to assemble data to the desired result, and 23.5% stated that they have problems understanding the meaning of the content of tables and columns. Only 14.7% stated that they do not have problems with the above mentioned aspects. 29.4% of the participants declared that they are not always able to access all desired data using queries as existing queries do not suit their individual needs and they are not able to create suitable queries by themselves. As a consequence, 44.1% state, that they use queries for their individual needs that have been created for them by others. We asked the participants not using queries, if they would use queries if they were better enabled to create queries for their individual purposes. 52.1% of these participants rated their intent on a scale ranging from 0 (no, that wouldn't provide any benefit to me) to 5 (yes, that would help me a lot) to be at least 3.

In accordance to our qualitative preliminary studies, using queries seems not to be an unproblematic issue for the participants. Nearly a third of the participants do not use queries, whereas more than half of these participants would use queries if it was easier to create queries suitable for their individual needs. Amongst the participants using queries nearly a third rate query creation and modification to be difficult, whereas the most problematic issue is finding needed data. Many participants are not able to create queries for their needs by themselves and rely on others creating queries for their needs.

3.5 EUD of Enterprise Queries

As identified in the previous section, the creation of custom queries to extract relevant data from enterprise systems is difficult for several reasons, like finding and understanding data in the complex data models of enterprise systems, or assembling data to the desired result by defining the according query itself. To improve information self-service capabilities of end-users, we developed an ontology-based architecture and EUD tool called "Semantic Query Designer" (SQD), enabling easy data access and query creation for end-users. The approach is based on a semantic middleware integrating data from one or multiple heterogeneous information systems and providing a simplified global data model in the form of a business level ontology (BO) that is comprehensible for end-users. SQD enables convenient visual navigation and query building upon the BO. Using SQD end-users may search for single concepts like "customer" in the data model, visually navigating alongside the relations to other concepts (e.g. following the relation "has address" to the concept "Address") or search for whole chains of relations in the data model, like in which ways "customer" is (directly or indirectly) related to "product". Desired concepts and relations can be added to a query using a single click. During navigation and query creation SQD provides a live preview of concept data and query results. All changes create immediate effects, increasing the confidence of the end-user in his or her design decisions. The defined query can either be deployed as a data source in the WCP and integrated in a custom widget (cf. Section 3.3) or exported to Excel for post-processing the data as desired. For a detailed description of SQD we refer to Spahn et al. [17].

As the SQD prototype was not ready for evaluation at the time of conducting the survey, we conducted a usability study to assess perceived usefulness and perceived ease of use. Nine ERP users from the already mentioned companies participated in the usability study, where they had to do several exercises in navigating the data model

and creating queries using SQD to get hands-on experiences and an impression of its capabilities. The exercises were videotaped and the opinion of participants assessed using a questionnaire and accompanying interviews. As a detailed discussion of the results would go far beyond the scope of this paper, we focus on few aspects in the following. Participants rated the perceived ease of use on a scale ranging from 0 (very hard to use) to 5 (very easy to use) with an average of 4.11. Asked how easy it is to learn the usage of SQD, participants rated the ease of learning with an average of 4.44 on a scale ranging from 0 (very hard to learn) to 5 (very easy to learn). In the accompanying interviews we asked participants if they would like to use SQD for their work. 8 out of 9 participants (88.9%) declared that they would like to use SQD. Asked if they think SQD could ease or speed up their work, also 8 out of 9 participants declared that they think SQD provides such benefits. In summary the results indicate a high perceived usefulness and ease of use, which are the main factors of acceptance according to the TAM [10].

4 Summary and Outlook

In this paper, we analyzed problem domains relevant for IM based on a questionnaire-based study in the context of three German midsized companies using ERP systems. With regard to the work context of end-users and their attitudes towards EUD, the results of our studies revealed that end-users mainly use ERP systems and office applications to accomplish their working tasks. It is common for end-users to create own solutions for their individual needs. They are willing to perform EUD activities and perceive such activities as being useful and beneficial. This positive attitude towards EUD can be seen as a positive indicator for the adoption of new EUD tools and methods.

We further investigated the potential of adopting EUD tools in the domains of information access and the flexible post-processing of enterprise data. Therefore we assessed the importance of the respective domain for the work of end-users, perceived pain points, the willingness to engage in related EUD activities and the perceived usefulness of EUD approaches addressing the respective domain.

With regard to the domain of information access, the results reveal that the work of end-users depends to a high extent on accessing data within enterprise systems using their respective GUI. End-users have to retrieve the same information recurrently and often in a stepwise manner, which is perceived to be cumbersome and results in the wish for easier access to information. Applications that are tailored to their needs and present all needed data for work tasks at a glance are perceived to be useful, and end-users declare willingness to invest effort in the creation of such solutions. We presented the WCP prototype to end-users, enabling them to create custom GUIs for information access in the form of widgets using a lightweight mashup paradigm. The rating of end-users revealed a high degree of perceived usefulness, perceived ease of use, as well as positive attitudes towards using WCP, which are the major factors affecting technology acceptance according to the TAM.

With regard to the domain of post-processing enterprise data, the results reveal that Excel can be considered to be a very important tool for the end-users, used in a significant amount of work time to post-process data from enterprise systems. Transferring data from enterprise systems is not always seamless, and the majority of

end-users would like to be able to transfer more or different data from enterprise systems to Excel, thereby easing or speeding up their work. At this, using queries seems not to be an unproblematic issue for end-users. Amongst the end-users using queries, many rate query creation and modification to be difficult, whereas the most problematic issue is finding needed data. Many participants are not able to create queries for their needs by themselves and rely on others creating queries for their needs. Many end-users not using queries would use queries if it was easier to create queries suitable for their individual needs. In a usability workshop we provided end-users with hands-on experience with regard to the SQD prototype, a visual query designer providing a simplified global data model comprehensible for end-users and based on an ontology-based architecture. The feedback of end-users indicates a high perceived usefulness, ease of use, and positive attitudes towards usage, indicating a high potential of being accepted according to the TAM.

Overall, we identified two domains that are relevant for IM that could be improved by the adoption of EUD tools and methods, as they are important to end-users and related EUD approaches are perceived as useful. We briefly described and referred to concrete prototypes we developed for addressing each of these domains. Although the feedback of end-users seems promising, further research and field studies in real enterprise contexts are needed to assess the applicability of the approaches and their acceptance more precisely.

Acknowledgements. We would like to thank Markus Wiemann for his support in conducting the discussed online questionnaire. The presented research was funded by the German Federal Ministry of Education and Research (BMBF) under the project EUDISMES (number 01ISE03C).

References

1. Krcmar, H.: Informationsmanagement. Springer, Heidelberg (2006)
2. Pfeiffer, S., Ritter, T., Treske, E.: Work Based Usability: Produktionsmitarbeiter gestalten ERP-Systeme “von unten”. Eine Handreichung. ISF (2008)
3. Spahn, M., Dörner, C., Wulf, V.: End User Development of Information Artifacts: A Design Challenge for Enterprise Systems. In: 16th European Conference on Information Systems (ECIS 2008), pp. 482–493. CISC (2008)
4. Spahn, M., Dörner, C., Wulf, V.: End User Development: Approaches towards a flexible Software Design. In: 16th European Conference on Information Systems (ECIS 2008), pp. 303–314. CISC (2008)
5. Gallivan, M.J., Keil, M.: The User-Developer Communication Process: A critical Case Study. Information Systems Journal 13(1), 37–68 (2003)
6. Stiemerling, O., Kahler, H., Wulf, V.: How to make Software softer - Designing tailorable Applications. In: 2nd Conference on Designing Interactive Systems (DIS 1997), pp. 365–376. ACM, New York (1997)
7. Beringer, J.: Reducing Expertise Tension. Commun. ACM 47(9), 39–40 (2004)
8. Lieberman, H., Paternò, F., Wulf, V.: End User Development. Springer, Heidelberg (2006)

9. Mehanjiev, N., Stoitsev, T., Grebner, O., Scheidl, S., Riss, U.: End-User Development for Task Management: Survey of Attitudes and Practices. In: 2008 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC 2008), pp. 166–174. IEEE Computer Society, Los Alamitos (2008)
10. Davis, F.D.: Perceived Usefulness, perceived Ease of Use, and User Acceptance of Information Technology. MIS Quarterly 13(3), 319–340 (1989)
11. Fishbein, M., Ajzen, I.: Belief, Attitude, Intention, and Behavior: An Introduction to Theory and Research. Addison-Wesley, Reading (1975)
12. Ajzen, I.: From Intentions to Actions: A Theory of planned Behavior. In: Action Control: From Cognition to Behavior, pp. 11–39. Springer, Heidelberg (1985)
13. Compeau, D.R., Higgins, C.A., Huff, S.L.: Social Cognitive Theory and Individual Reactions to Computing Technology: A Longitudinal Study. MIS Quarterly 23(2), 145–158 (1999)
14. Kvale, S.: Interviews: An Introduction to qualitative Research Interviewing. Sage Publications, Thousand Oaks (1996)
15. Spahn, M., Wulf, V.: End-User Development of Enterprise Widgets. In: 2nd Int. Symposium on End User Development (IS-EUD 2009). LNCS, vol. 5435, pp. 106–125. Springer, Heidelberg (2009)
16. Davis, F.D.: A Technology Acceptance Model for empirically testing new End-User Information Systems: Theory and Results. PhD thesis, Sloan School of Management, Massachusetts Institute of Technology, Cambridge, MA, USA (1986)
17. Spahn, M., Kleb, J., Grimm, S., Scheidl, S.: Supporting Business Intelligence by Providing Ontology-Based End-User Information Self-Service. In: 1st Int. Workshop on Ontology-supported Business Intelligence (OBI 2008). ACM, New York (2008)

Evaluating the Accessibility of Websites to Define Indicators in Service Level Agreements

Sinésio Teles de Lima[1], Fernanda Lima[2], and Káthia Marçal de Oliveira[1]

[1] Catholic University of Brasilia
SGAN 916 Norte AV. W5, 70.790-160, Brasília, DF, Brazil
stlima@gmail.com, kathia@ucb.br
[2] Computer Science Department, University of Brasilia
Campus Universitário Darcy Ribeiro, ICC Centro, Asa Norte
Caixa Postal 4466, 70910-900, Brasília, DF, Brazil
ferlima@cic.unb.br

Abstract. Despite the evolution of the Internet in the past years, people with disabilities still encounter obstacles to accessibility that impede adequate understanding of website content. Considering that Web accessibility is an added value to the website, it is important to have in place monitoring mechanisms and website accessibility controls. Service level agreements (SLA) can be used for this purpose, as they establish, by means of a service catalog, measurable indicators that certify the fulfillment of preset goals. This paper proposes a way to evaluate the accessibility of websites through a practical approach utilizing software measures, with the purpose of collecting data to define indicators for a service catalog of an SLA of website accessibility. Initial application of the approach was conducted on Brazilian federal government websites with the participation of ten users with visual disabilities. The study shows the viability of defining indicators.

Keywords: Accessibility, Software measures, Service Level Agreement.

1 Introduction

Organizations have turned to Web technology support to increase their exposure, to attract potential clients and retain existing ones, and to facilitate contact with partners and suppliers. Nonetheless, the available content on most websites worldwide presents restrictions for users with visual, auditory, or neurological disabilities, among other impediments.

Website managers should provide content with adequate levels of accessibility so that people with disabilities can interact properly with the Web; that is, Web accessibility must be seen as a service to be made available to a community of users. Moreover, it is important that Web accessibility be measured and evaluated for conformity with rules of accessibility.

Service level agreements [12] are instruments that can be used to effectively monitor levels of service related to accessibility. An important component of SLA is the service catalog, which describes the services and indicators needed to monitor the fulfillment of the SLA.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 858–869, 2009.
© Springer-Verlag Berlin Heidelberg 2009

This article presents an evaluation approach of website accessibility and defines the indicators required to compose a service catalog of SLAs.

In the following sections, concepts regarding Web accessibility (section 2) and service level agreements (section 3) are presented. Next, the following are presented: our evaluation approach (section 4), its application (section 5), related studies (section 6), and conclusions of this study (section 7).

2 WEB Accessibility

Web accessibility is the degree to which people with visual, auditory, physical, speech, cognitive, or neurological disabilities can perceive, understand, navigate, and interact with the Web [7].

The Web Content Accessibility Guidelines 1.0 [4] are composed of 14 general guidelines organized into checkpoints that address specific aspects of accessibility. Each checkpoint is rated according to the criticality of implementation of the accessible content (that is, when a content developer must [priority 1] or should [priority 2] satisfy the checkpoint; or when s/he may address the checkpoint [priority 3]).

In WCAG version 2.0, currently with "Proposed Recommendation" status, checkpoints and priorities are referenced as success criteria and conformance level, respectively. It also defines four basic principles: the content must be *perceivable*, the interface components must be *operable*, the content and controls must be *understandable*, and the content must be sufficiently *robust* to interact with the user agents. Each principle contains general guidelines that are organized into conformance levels and success criteria [3].

To conciliate the innovations proposed by the new version, the W3Consortium (W3C) has attempted to define mappings between the two versions, defining a correlation from the principles, guidelines, conformance levels, and success criteria of version 2.0 with the guidelines, priorities, and checkpoints of version 1.0 [2].

3 Service Level Agreement

Service level agreements (SLAs) are formal agreements that aim to establish quality standards for services rendered by providers to their clients [12]. Using an SLA, clients can demand the fulfillment of quality specifications required for products and services acquired from their providers [13]. In general, an SLA contains the participants of the SLA (providers and clients) and their respective responsibilities; efficiencies, service catalog, procedures for collecting data, form of payment, and fees and incentives [13, 17, 15, 11].

The creation of a service catalog represents the most critical phase of SLA, for it is in this phase that the client's concerns are converted into service level objectives. The service catalog defines (1) the services that compose the scope of the SLA, (2) a detailed description of each service, (3) the measurable attributes of the characteristics of the services, and (4) the steps needed to confirm whether the expected service level was achieved in each service rendered [13].

4 Approach to Defining SLA Accessibility Indicators

The first step toward defining the indicators of a service catalog is the identification of what to evaluate with regards to accessibility. For this purpose, the Goal Question Metric [1] was used, which is based on the premise that in order to measure something it is necessary to define the objectives (goal), the key questions described in the objectives (question), and finally the steps that permit the evaluation of the fulfillment of the objectives (metric). The objective was established as follows:

Analyze content on websites,
with the purpose of evaluation,
with respect to accessibility,
From the viewpoint of the user with visual disabilities (in order to limit the scope).

Starting with the definition of accessibility and its principles, presented in section 2, the following questions arose:

- Q1: What is the degree of perception of the content of the websites evaluated?
- Q2: What is the degree of operability of the content of the websites evaluated?
- Q3: What is the degree of understanding of the content of the websites evaluated?

Meanwhile, two other aspects are considered important: the adequate completion of an interactive activity on the Web and the time spent by the user during the interaction with the Web. To define questions related to these aspects, the portion of the ISO/IEC 9126-4 (2001) norm concerning quality in use was investigated, given that the focus of the evaluation is the accessibility of sites by users.

Quality in use [9] refers to a software product's efficiency, productivity, satisfaction, and security. *Security* refers to the levels of risk of physical damage to the user or the risk of economic damage caused by the use of the software products, and therefore this classification does not apply to this study. *Productivity* refers to the capacity of the software product to facilitate the employment of an appropriate quantity of resources that ensure high efficiency rates, which are then related to users' capacity to reach their specified goals with accuracy and thoroughness. *Satisfaction* refers to the capacity of the product to satisfy users. Based on these concepts the following questions were elaborated:

- Q4: What is the degree of productivity of users during their interaction on the Web?
- Q5: What is the degree of satisfaction of users when executing tasks?

To define the measurements regarding these five questions (Table 1), the frequency of checkpoint violations, which refer to the accessibility principles of the WCAG, and the ISO 9126-4 (2001) measurements, which refer to the quality during use, were considered.

Objective methods (based on numerical rules) and subjective methods (based on user opinion) [8] were used to collect the measurements. The objective measurements were done with tools that collect data concerning real and potential violations, as well as the principles of accessibility (resulting in measurements M2, M7, and M9).

The subjective measurements were registered in a collection protocol based on participant observation methods [10] consisting of evaluation sessions with disabled users in which an observer interacts with the user to provide him/her with basic information and an accurate understanding of the objective for each task, thereby establishing a dialogue for registering difficulties and strategies adopted by the user during task execution. After completion of the sessions, the user's perception with regard to the accessibility of the site was registered (measurements M1, M3, M4, M5, M6, M8, and M11) along with the time spent to complete each task (M10).

Finally, to complete the evaluation one must:

- **Choose a tool** that supports the Web navigator in use, allowing the evaluation of single pages in a specific Web domain (intra-domain analysis) and multilevel evaluation of each site.
- **Choose a website** that is considered to have the broadest array of services and that is most relevant and interesting to users participating in this study.
- **Choose tasks** relative to the evaluation of each site.
- **Plan sessions** of evaluation with respect to user choice, evaluation locale, configuration of the necessary resources for the evaluation (including technologies for assistance), and scheduling and definition of time necessary for each evaluation.
- **Execute** evaluation sessions according to observation/participation method.
- **Collect and analyze the data** using the collection protocol and the chosen tool.
- **Relate data and define a services catalog** - the data collected using objective and subjective methods are related in the search in order to identify trends that indicate relationships between frequency of violations and other measurements, such as the time spent performing a task and the levels (degrees) of perception of the user as to how they perceive, operate, and understand the contents. Additionally, it is important to look for relationships between the degree of satisfaction and the frequency of violations.

5 Application

Motivated by Decree 5,296, which makes compliancy to the rules of Web accessibility and efficiency mandatory for Brazilian government websites, the initial application of this approach was applied to the context of federal government sites.

The tool chosen was the TAW Standalone (http://www.tawdis.net/taw3/cms/en) — since it is a free tool and adhering to the previously defined criteria.

The choice of sites was made based on information published on the iBest Awards site (edition 2006), which lists the best websites on the Brazilian Internet awards circuit. The category of citizenship and subcategory of government were considered which include websites that render government services—the following sites having received the top ratings: Secretaria da Receita Federal (Secretary of the Internal Revenue), Previdência Social (Social Security), and Instituto Brasileiro de Geografia e Estatística – IBGE (Brazilian Institute of Geography and Statistics).

Table 1. Accessibility evaluation measures

Q.	Measure	Description	Scale
Q1	M1. Degree of perception of content.	Measures the user's perception regarding Web content read by a software screen reader (Jaws, Virtual Vision etc.).	From 0 (totally incomprehensible) to 7 (totally comprehensible)
	M2. Number of violations of the Perception principle.	Measures the number of violations found on Web pages pertaining to a task and related to items in WCAG 1.0.	Integer number (X >= 0)
Q2	M3. Degree of operation of content with use of the keyboard.	Measures user perception regarding the operation of Web content with the use of a keyboard or directional instrument (mouse).	From 0 (totally inoperable) to 7 (totally operable)
	M4. Degree of operation of content in relation to the time of execution.	Measures user perception regarding the time spent during interaction with Web content.	From 0 (totally unsatisfactory) to 7 (totally satisfactory)
	M5. Degree of operation of content in relation to the complexity of the navigation.	Measures user perception regarding the complexity of navigating during an interaction with Web content.	From 0 (very complex) to 7 (very simple)
	M6. Degree of operation of content in relation to the existence of anchors.	Measures user perception regarding the ease with which users are able to find anchors during the interaction with Web content.	From 0 (very difficult) to 7 (very easy)
	M7. Number of violations of the Operation principle for the task.	Measures the number of violations found on Web pages related to the items that compose the Operation principle.	Integer number (X >= 0)
Q3	M8. Degree of Understanding.	Measures user perception with regard to understanding of Web content during the execution of a task.	From 0 (totally incomprehensible) to 7 (totally comprehensible)
	M9. Number of violations of the Understanding principle for the task.	Measures the number of violations found on Web pages related to items that compose the Understanding principle.	Integer number (X >= 0)
Q4	M10. Time of execution of the task [9].	Measures the time spent by the user to execute a task.	Time interval (X >= 0).
Q5	M11. Degree of satisfaction in the context of accessibility [9].	Measures the degree of user satisfaction in relation to the interaction with Web content in the context of accessibility.	From 0 (totally unsatisfied) to 7 (totally satisfied)

To minimize execution time, one task was chosen for each site. These were identified by the available services and required a degree of complexity for the execution of the task. The tasks chosen were: (T1) consult an individual income tax return (http://www.receita.fazenda.gov.br), (T2) register for Social Security (http://www.previdência. gov.br), and (T3) contact IBGE (www.ibge.gov.br).

To plan the evaluation session it was established that the participanting users must have different levels of visual disability and experience using the Web, so that the greatest representation of data collected would be guaranteed. Eighteen people were contacted, and ten agreed to participate as volunteers in the evaluations. The evaluations were performed in the homes of the users for their convenience as well as for the convenience of providing the necessary technical requirements, such as screen reader software (Jaws or Virtual Vision), for the evaluations.

The evaluation sessions were recorded on video with previous consent from the users and were initiated with explanations of the evaluation and the collection protocol, along with verification that the technical items were functioning adequately. The average time for each session was two and a half hours.

The chosen tool reported the frequency of violations, real and potential, for each verification checkpoint. Using the mapping of the WCAG 1.0 verification items as the success criteria of WCAG 2.0 [2], the data for each accessibility principle (that is, related to questions 1, 2, and 3) were obtained as shown in Table 2.

Table 2. Number of violations by principle

Task	Question (Principle)	Real Violations	Potential Violations
T1	Q1. (Perceivable)	23	151
	Q2. (Operable)	2	69
	Q3. (Understandable)	5	36
T2	Q1. (Perceivable)	59	445
	Q2. (Operable)	0	58
	Q3. (Understandable)	4	19
T3	Q1. (Perceivable)	22	59
	Q2. (Operable)	0	53
	Q3. (Understandable)	0	38

The demographic data shows that 60% of the users have completed or are in the final stages of completion of their undergraduate degrees; 80% have been active in the job market for more than five years; 100% use the Web to study or work; and 70% use the Web for more than 24 hours weekly. In the context of visual disabilities, 30%, 20%, and 40% of the users have average levels, severe, and total visual disability, respectively. In relation to their experience with the Web, 80% have worked with the Web for more than five years.

With regard to the observation data, all the users managed to complete tasks T2 and T3, however just one completed task T1. The execution times varied from 6 to 28 minutes for task T2, and 5 to 18 minutes for task T3.

Table 3 shows evaluation data of users regarding the items for each question. This table presents the average value calculated, considering the evaluation of the ten users and the average value for the question.

Table 3. Evaluation data of 10 users

Task	Question	Measure	Measure Average	Question Average
T1	Q1 (Perceivable)	M1	2.90	2.90
	Q2 (Operable)	M3	5.40	4.48
		M4	2.30	
		M5	5.00	
		M6	5.20	
	Q3 (Understandable)	M8	3.30	3.30
	Q5 (Satisfactory)	M11	1.00	1.00
T2	Q1 (Perceivable)	M1	5.50	5.50
	Q2 (Operable)	M3	5.80	5.68
		M4	5.10	
		M5	5.80	
		M6	6.00	
	Q3 (Understandable)	M8	5.60	5.60
	Q5 (Satisfactory)	M11	5.60	5.60
T3	Q1 (Perceivable)	M1	6.00	6.00
	Q2 (Operable)	M3	6.20	5.95
		M4	5.80	
		M5	6.00	
		M6	5.20	
	Q3 (Understandable)	M8	3.30	6.00
	Q5 (Satisfactory)	M11	1.00	1.0

The following analysis of the data was derived from the data collected:

- Users, in general, obtained a high degree of perception of the content of tasks T2 and T3, observing that the average result was at almost the maximum scale value (7). For task T1, the degree of user perception was considerably lower, indicating that the content of this task was not perceived satisfactorily by the users. This can be explained as due to the use of distorted images (captcha) and the lack of equivalent textual or auditory info on the specified page that composed task T1.
- Tasks T1 and T3 obtained the lowest frequency of violations for the perception principle. Because of the user's low degree of content perception in task T1, it was anticipated that the quantity of violations for task T1 would be the highest among the three tasks. This condition is explained by the fact that the evaluation tools do not yet correctly identify the use of distorted images without equivalent textual or auditory information and, in this case, stricter rules regarding the identification of these obstacles to accessibility should be applied. Although task T1 had only one distorted image, it is evident that the negative impact on the user in relation to the Web content points to the need for a textual or auditory equivalent in these cases.
- In general, all content operation tasks can be considered to be in the satisfactory range, with the exception of task T1, which only one user completed and obtained a degree of operation of content below the one recorded for the other tasks.

- One can observe that the higher the level of content operation, the lower the number of violations found for this principle per task.
- The results obtained for the degree of content comprehension per task indicate that distorted images also affect content comprehension, considering that task T1 values scored much lower when compared to the values obtained by the other tasks.
- Notably, the higher the levels of comprehension of task content, the lower the quantity of violations found for that principle.
- Execution times registered for task T2 and T3 were sufficiently satisfactory, as 80% of the users, in both cases, managed to conclude this task in less time or equal to the average execution time for each task. On the other hand, just 10% of the users managed to complete task T1 (a user with average level of visual disability).
- It was observed that the percentage of users who completed the tasks in less time or equal to the average time spent per task, was also registered as directly corresponding with rates of degrees of perception, operation, and content comprehension, as the attributed values of these principles grew while the percentages increased. This same relationship was observed with regard to satisfaction. Therefore, it can be said that user satisfaction is directly related to productivity.
- User satisfaction was also affected by the number of principle violations; that is, the higher the rate of violations, the lower the level (degree) of user satisfaction. Meanwhile, this fact was observed for the number of violations related to the operation and comprehension principles. For the principle of perception, this relationship was confirmed in task T1 and T3 only.
- In contrast, the degree of user satisfaction has a direct relation to the measure of execution time. The degree of satisfaction is greater than the average of the percentage of increase in the completion rate. The same ratio was observed for the percentage of users that completed the task in less or equal time to the average time recorded for each task. This indicates that user satisfaction is in direct relation to the completion of the tasks and the time spent by the users on a particular activity.

Finally, to relate the data and define the indicators in a service catalog, it was necessary to refine the data of items violated by the accessibility principle by reconsidering the mapping of version 1.0 with 2.0. For this analysis, items with checkpoint 12.4 were not considered (priority 2). These items refer to the use of HTML labels and their respective attributes to improve accessibility. Items that obtained percentages of violations equal to null in all of the occurrences, considering one or more tasks, were also not considered. In this way, seven items were found important to be evaluated. In this group:

- Four items presented an inverse relationship between the percentage of violations by task and the average of the evaluation completed by users; that is, the larger the percentage of violations the less the average of evaluation per principle obtained for these items up to a minimum of 2 distinct tasks (being 57% for 3 tasks).
- Three items presented exceptions to the inverse relationship observed (each in a single instance).

Based on this situation, it was decided to propose maximum percentages of violations to the items of accessibility. The criteria for establishing these percentages followed the goal of obtaining seven points for the measurement of the notes related to the principles of accessibility. In this context, the following sequence of steps was executed with the aim of obtaining the maximum percentages of violations for each item:

- Projection of the maximum percentage of violations for the four items (1.1 [priority 1], 3.2, 3.3, and 7.3 [priority of 2]) that proved the inverse ratio for the three tasks.
- Calculation of the maximum percentage of violations using cross-multiplication for inversely proportional values, considering the following question for each individual item per task: What is the percentage of violations for which the average degree per principle is equal to seven (7)?
- Choice of the lowest percentage among the three tasks as a maximum percentage tolerated per item.

Table 4 presents the maximum percentages of violations calculated per task—the last column presenting the maximum tolerance per item.

Table 4. Maximum percentages tolerated by task and per item

Question (Principle)	P	Item	% Maximum by task			% Max
			T1	T2	T3	per item
Q1. (Perceivable)	1	1.1	14.1	20.5	22.1	14.1
	2	3.3	2.0	0.4	0.0	0.0
Q2. (Operable)	2	7.3	21.3	0.0	0.0	0.0
Q3. (Understandable)	2	3.2	28.3	0.0	0.0	0.0

The verification item 1.1, being priority 1, has the greatest relevance in the context of Web accessibility when compared to the other items. This item previews the use of textual equivalents for each non-textual component of a Web page. It also aims to serve the basic necessities of the user with visual disabilities as well as the perception of Web content.

In the registers done on video, it was observed that the users are completely dependent on the screen readers to correctly perceive the Web content. The screen readers, in turn, are dependent on the textual equivalents so that they can inform the correct description of the non-textual components (images, anchors, tables, etc.). The absence of textual equivalents causes the screen reader to ignore non-textual elements or to transfer insignificant information to the user, making it impossible for users to interact transparently with the Web content.

In one session registered on video, one of the users, after having reached the objective of task T2, wanted to advance further and tried to begin the process of registering for Social Security. With this, the links related to the registration and the instructions for the registration were represented in images that did not have appropriate textual equivalents. The user tried various strategies of interaction in the hope that the screen reader would inform about the content, ultimately without success. In the end, the user considered the task of "Registering for Social Security" frustrating and inaccessible.

Even though the lowest percentage value of violations for this item of verification is 14.15%, the absence of textual correspondents affects overall perception of the Web content. Therefore, the target percentage for the catalog of services related to this item should be 0% as a prerogative to guarantee accessibility.

The maximum percentages tolerated obtained for the most significant item, which refer to violations of the items of verification and principles of Web accessibility, are the quality indicators that compose the service catalog of SLA for Web accessibility. Considering the results obtained, it is correct to affirm that the service level agreements for Web accessibility can require that the percentages of violations of the items of accessibility be the lowest possible, reaching zero, so that the perception, operation, and comprehension of the content is adequate for all users with disabilities.

In this context, the service catalog of SLAs for accessibility can contain explicit clauses that define the maximum percentages of violations tolerated and the desired percentages for each item of verification or success criteria. In Table 5 the indicators (percentage of violations) are presented for each item identified as critical, after the analysis of collected data. These indicators can compose catalogs of future services of SLAs for Web accessibility among clients and service providers that involve the development of websites.

Table 5. Indicators for a service catalog of SLA for Web accessibility

WCAG 1.0		WCAG 2.0			% maximum violations tolerated	% of desired violations
Priority	Checkpoint	Principle	Level	Success criteria		
1	1.1	Perceivable	1	1.1.1	14.2%	0.0%
2	3.3	Perceivable	1	1.3.3	0.0%	0.0%
2	7.3	Operable	2	2.2.3	0.0%	0.0%
2	3.2	Understandable	1	4.1.1	0.0%	0.0%

Summary of item descriptions:

- 1.1 (priority 1): Use textual equivalents for all of the non-textual components pertaining to all the pages of the website.
- 3.3 (priority2): Use stylized forms to control the format and the presentation of content.
- 7.3 (priority 2): Avoid content movement, in the case of the user agents not being able to support the freezing of this function.
- 3.2 (priority 2): Create documents that are valid and coherent with formally established grammar.

6 Related Work

Various authors have proposals for measuring accessibility. Some of the more relevant are:

- [16] propose a measurement of the potential problems in order to establish a value referring for the number of obstacles found in relation to the number of potential obstacles on a site.
- [6] propose a measurement to obtain an accessibility index for blind users calculated considering the guidelines defined by WCAG, and that a Web page is accessible if all the elements are accessible.

- [14] also define a Web accessibility rate (WAB - Web Accessibility Barrier) considering potential problems, page numbers, and weight, among other aspects.
- [5] proposes to add accessibility test results.

All of these studies, however, consider the definition of an objective measure when evaluating accessibility, considering different factors. The approach in the present study seeks not to only collect objective measurements, but to compare them with subjective ones to define indicators for an SLA of accessibility.

7 Conclusions

This study presented a practical approach to the evaluation of website accessibility that involves subjective evaluation by users with disabilities in conjunction with objective data collected using specific tools. This approach, besides serving to analyze the accessibility of websites, facilitates the definition of service catalog indicators of SLAs for Web accessibility established between clients and Web development providers or even composes specific sessions of accessibility for SLA generics that involve the development of websites.

Future studies intend to define new indicators for the service catalogs, suggesting a broader range of evaluators and a greater number of sites and tasks to evaluate. There are plans to use tools developed for the 2.0 version to collect the number of violations.

Acknowledgements. The authors thank the users who participated in the application and the CNPq, an entity of the Brazilian government dedicated to scientific and technological development.

References

1. Basili, V.R., Caldiera, G., Rombach, H.D.: The Goal Question Metric Paradigm. In: Encyclopedia of Software Engineering, vol. 1, pp. 528–532. Wiley, Chichester (1994)
2. Caldwell, B., et al. (eds.): Comparison of WCAG 1.0 checkpoints to WCAG 2.0 (2006), http://www.w3.org/TR/2006/WD-WCAG20-20060427/appendixD.html (accessed on April 07, 2006)
3. Caldwell, B., et al. (eds.) Web Content Accessibility Guidelines 2.0: W3C Candidate Recommendation, April 30 (2008), http://www.w3.org/TR/WCAG20/ (accessed on May 10, 2008)
4. Chisholm, W., Vanderheiden, G., Jacobs, I.: Web Content Accessibility Guidelines 1.0 (1999), http://www.w3.org/TR/WCAG10 (accessed on April 07, 2007)
5. Cluster, W.: W.A.B. Unified Web Evaluation Methodology (UWEM 1.0), http://www.wabcluster.org/uwem/ (accessed, January 2007)
6. González, J., Macías, M., Rodríguez, R., Sánchez, F.: Accessibility Metrics of Web Pages for Blind Endusers. In: Proceedings of 2003 International Conference on Web Engineering, Oviedo, pp. 374–383 (2003)
7. Henry, S.L., et al. (eds.), Introduction to Web Accessibility (2005), http://www.w3.org/WAI/intro/accessibility.php (accessed on October 10, 2006)

8. ISO/IEC 15939, Software engineering – Software measurement process (2002)
9. ISO/IEC 9126-4, Software Engineering - Product quality – Part 4: Quality in Use Metrics (2001)
10. Melo, A.M., Baranauskas, M.C.C., Bonilha, F.F.G.: Avaliação de Acessibilidade na Web com a Participação do Usuário: Um Estudo de Caso. In: Proceedings of the Brazilian Human-Computer Interaction Symposium (2004)
11. Morrison, M.: Computer Resources Core Services Catalogue. Version 2.0 (2006), http://www.myitiltemplates.com/View.php?templateID=6&c=7&q (accessed on 12 August 2007)
12. Muller, N.J.: Managing Service Level Agreements. International Journal of Network Management 9(3), 155–156 (1999)
13. Office for Government Commerce (OCG), ITIL – The Key to Manage IT Services: Service Delivery – Version 1.2; Crow (2001)
14. Parmanto, B., Zeng, X.: Metric for Web Accessibility Evaluation. Journal of the American Society for Information Science and Technology 56(33), 1394–1404 (2005)
15. Sturm, R., Morris, W., Jander, M.: Foundations of Service Level Management. Indianapolis. Sams (2000) (2001 in text)
16. Sullivan, T., Matson, R.: Barriers to use: Usability and Content Accessibility on the Web's Most Popular Sites. In: Proceedings on the 2000 conference on Universal Usability, pp. 139–144. ACM Press, New York (2000)
17. Walker, G.: IT Problem Management. Prentice-Hall, Englewood Cliffs (2001)

Promoting Collaboration through a Culturally Contextualized Narrative Game

Marcos Alexandre Rose Silva and Junia Coutinho Anacleto

Federal University of São Carlos, Washigton Luis KM 235, São Carlos, São Paulo, Brazil
{marcos_silva,junia}@dc.ufscar.br

Abstract. This paper describes a research about developing web narrative game to be used at school by teachers, considering students' culture expressed in common sense knowledge, for storytelling, allowing teacher to create stories according to student's cultural reality, and consequently enabling students to identify and get interested in collaborating with the teacher and other students to develop the story, being co-authors. Therefore this game can allow students to learn how to express themselves, which means to leave their imagination flow and to allow them to adequately understand and elaborate situations experienced in school, family and community with no impact on their real life. Through stories students also can learn how to work collaboratively, to help, and to be helped by their friends and teacher. This context is also applicable on companies, considering teamwork and the necessary role each one has to play for collaborative work.

Keywords: Collaboration, Storyteller, Narrative Game, Context, Common Sense, Education, Educational game.

1 Introduction

Knowing how to cooperate, to negotiate, to express; in fact, to work in group is a very important issue in business world. Therefore, people should learn these skills since they are children; however, in Brazil and in other emergent countries teaching those skills at school can still a challenge. These abilities are also part of fundamental educational objectives because when children participate actively in their class, they cooperate with the teacher and other students, building their own knowledge [4].

Another skill that is important in business and education is to know how to live and to communicate with different people because each person has his own culture, values and socio-cultural reality. Because of that, working or studying with different people is challenging. Activities to promote work in group can rarely occur spontaneously [4], so teachers and students need to have activities and tools supporting this new way of studying. Therefore, it is presented here an educational computer narrative game to support teachers so that they can work collaboratively with their students through storyteller. This context can be changed for another like workspace and teamwork in companies, considering the necessary role each person has to play for collaborative work. So it can be considered as an environment to test people's skills on how to deal

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 870–881, 2009.
© Springer-Verlag Berlin Heidelberg 2009

with other colleagues, how to react in certain situations and how to collaborate with each other to solve conflicts and work collaboratively.

This game is inspired in Role-Playing Game - RPG [2]. In this type of game there are participants, the master, who usually is the most experienced player and his task is to present the story to a group, with characters, their characteristics, scenarios; in short, the necessary descriptions to compose an adventure with puzzles, situations and conflicts that require choices by other participants, who are the players.

These players are not just spectators; they contribute actively in the story, through their characters that choose paths and take own decisions, and most of the time not foreseeing by the master, contributing to the spontaneous and unexpected development of the story. The master can interfere in the narration, describing the scenarios, the characteristics, the objects that appear in the narrative environment and proposing situations so that the characters can interact. In the context work the master is the teacher who introduces the story and intervenes collaboratively with the players. The players are the students, the co-authors of the narrative.

The master defines a common objective to the group. For instance, he provides anything so that the students can guess the end of the story. Therefore, each person through the character needs to collaborate with each other to achieve the objective. Nevertheless, students need to identify and get interested in collaborating developing the story. Therefore, this narrative game allows teacher to create stories according to the student's social economics and cultural reality, taking into consideration the students' culture expressed in common sense knowledge. Because of that, teachers use a vocabulary that is familiar to the students, considering their myths, beliefs, taboos and knowledge.

Students have the opportunity to be closer to the contextualized stories, and consequently it allows them to participate and express themselves. Teachers also can monitor the children's learning process through the stories, and teachers are able to support them and intervene whenever necessary, promoting a safe and healthy student's development.

Collaborative storytelling enables people to be attentive and interested in what is happening in the story in order to understand it and consequently to contribute to it. That attention and concern to understand what other people say allow them to the come closer because they are interested in what the other person has to say, even indirectly [10].

Fantasy in narrative games allow people, especially children, to feel safe to express themselves, to talk about situations that occur in their lives because they believe that what happens in fantasy has little or even no consequence in real life. Therefore children often think that it is easier and safer to express themselves through characters; they feel less threatened to express hostility in the story because they express their emotions, joy, sadness, anger and euphoria through their characters that act in accordance to people's emotion.

According to Oaklander [10] children do things, behave and move in their fanciful world in the same way in their real world. Because of that narrative games for their free expression and support to try experiences are useful. Narrative games may help children to express, and the teacher to have the opportunity to observe the children's behaviour throughout the story, also permitting a genuine contact among themselves.

This game is been developing to children from 8 to 12 years old. According to Piaget [13] in this phase the children are in the stage called Operational Concrete

Thought. In this stage child has great interest in games and finds new ways to play and to work collaboratively. These are important features for a game that allows people to tell stories collaboratively.

In this phase children develop academic instruments such as reading, writing and basic math, and they are able to focus their attention. Thus, children have the capacity to read the story being told, to help to write it, i.e., to participate in building the story and to get attentive to the whole story [13].

During Operational Concrete children are willing to make friends and want to participate and interact with other children's game. Therefore, there are great chances that children can be interested in participating and interacting with the story being told collaboratively.

This paper is organized as follow: section 2 is about related works and the game's prototype is presented; section 3 describes the common sense knowledge use and collect; and section 4 presents some conclusions and future works.

2 Narrative Games

There are many narrative games that teachers use in the classroom, such as: Aulativa [8], Taltun [16], Revolution [15], among others. For example, Revolution is a game based on historical events of the American Revolution. This game teaches students about historical events that includes the daily social, economic, and political; Neverwinter is a game that is set in a huge medieval fantasy world and it allows students confront educational tasks and puzzles.

All these games have common characteristic, as a previously defined context. They have a fixed set of characters, scenarios and themes to the storytelling. Because of this, if teachers want to use these games, they need to adapt their classes to the games rules. They also do not consider student's culture, knowledge. Some researchers Papert [11], Vygotsky [17], Freire [6] have described that when children identify the relation between what they are learning and their reality, they feel themselves more interest. In short, they can identify that the semantic of the words is significant to their life, because it is close to their reality.

Because of this, the narrative game proposed in this paper, Contexteller is a storyteller environment contextualized by common sense knowledge and it intends to support teachers on telling stories collaboratively with students according their pedagogical objective.

2.1 Contexteller

The players perform, speak, think through their characters and decide the character's attitudes. In the narrative game described here, each player chooses a card provided by the master and each card represents a character. The card refers to a form of playing RPG, in which the players around of table present their cards to the other participants. Through the characters represented by these cards they build a story, from the scenario already defined by the master [5]. During the story, the master can interfere describing a new scenario and new situations so that the characters can decide how to act.



Fig. 1. Interface of the player's card

The interface of the card, shown in Fig. 1, is blue because more than half of the western population prefers this color [12]. According to Pastoreau [12] blue represents the infinite and dream, consequently this color can represent the imagination and fantasy existing in a narrative game.

The card has some RPG elements, such as: Magic, Force and Experience. The values of the first and second elements are defined by the players but they need to consider the overall scores previously set by the master. These elements are considered to be one of the rules existing in RPG. This rule avoids many discussions that could occur during the story. For example, knowing what is the strongest or most powerful character. The values of the elements are numbers to be considered in some situations. For example, a character with Force equal 5 is more likely to survive a crash than a character with Force equal 2.

Piaget [13] describes that children from 8 to 12 years old have Serial Sorting relationship because they can range from the highest to the lowest value but they have difficulty in understanding when something is presented in abstract form.

Therefore, this game uses numbers to represent the values of Power, Magic and Experience elements, and also uses plus and minus signs to change the value. Through the numbers, children can understand the values of those elements can increase or decrease and can compare it with the values to other players' elements.

The master attributes the value of the Experience when the character achieves a particular goal stipulated during the development of the story, in short, dynamically. This latter element stimulates the student to play carefully, to want to confront and to overcome the challenges.

The master can offer advantages to more experienced players, for example, when it is necessary to choose between two paths to be taken in a story. So, the teacher can allow the player, with the highest value of experience to choose between the two paths. This value can also add advantages to other the character's card elements.

Therefore, these three elements together with the master's narration provide competition to the game, even indirectly. According to Crawford [3] conflict is an important characteristic of the game. He reports that there is no game without conflict, even when there is not direct competition among the players.



Fig. 2. The interface of the Narrative Game

Fig. 2 shows the interface available for players. This interface allows the players to see their card (I), their dice (II), and the text area (III), which allows them to the master read all the messages sent to them and master during the composition of the collaborative story. In area (IV), the card, with another color and size, represents the master of the game, and area (V) shows the cards of other players.

The dice, which is part of the RPG, shows the players and the master whether a particular action is possible, or not [14]. For example, to raise any object it is necessary to have the value of the Force element equal or greater than the value of the object weight. This weight is defined by value of the dice thrown by the player. Thus, the player can raise the object if the value of the Force is equal to N and the value of the dice is from 1 to N. If the value of the dice exceeds N such action will not be possible because the value of the weight of the object is more than the value of the Force element.

Each player sees the master and all the other players, along with the values of the card elements and their respective images through the interface of the game. However, because of the need to understand the elements Magic, Force and Experience in an easy an agile way, it was chosen only the images that represent such elements in the others cards. The player always sees the same images with their names in his card.

All features available to the player in the interface of the game, such as the elements, the place where the player inserts the written text during the story, are visible and available on or near the card, to facilitate their location. According to Jarvinen et al., [7] an important feature for the game to be exciting is to permit the player to find all the features in the interface, required to perform a certain task, in 30 seconds.

Considering the narrative game in which the player must be aware of the story, he can not get dispersed for a long time to find the features, which must be close and easy to find.

The interface of this game has 6 cards, 5 players and 1 master. Number 6 is usually used in RPG of cards [5]. 5 players also facilitate the master (teacher) to monitor the whole story that is being told by the players (students). If the number of players was greater than 5, the master could face difficulties in reading all messages, in

interacting appropriately during the story and in observing the development and behaviour of each character.

This game intends to enable the teachers to get to know their students and the students how to express themselves and to work collaboratively. So teacher needs to be attentive to all messages, to interact and to instigate the students to continue the story. These features will allow the master:

- Having more attention to each player. According to Piaget [13] when somebody wants that the children learn the moral of some story, it is necessary spend some time talking to them about it;
- Knowing the way and often that the players contributed with the story. If master to observe a player avoiding to say something or disinterested in the story, the master can interact with the player. He can comment or ask about events of the story, in short, instigating the player to participate of the story.

To help teachers to create and to tell stories this game has as objective to give computer support to the master so that he can get help from contextualized information, both in the initial phase , i.e., the composing of scenario and characters to be presented and in other phases, such as: story definition and sequence. This support is obtained through common sense that represents cultural aspects of the students´ community.

2.1.1 Use of Common Sense Knowledge in the Narrative Game

The game proposed in this paper, uses the common sense knowledge obtained by the Open Mind Common Sense in Brazil Project (OMCS-Br), developed by the Advanced Interaction Laboratory (LIA) at UFSCar in collaboration with Media Lab from Massachusetts Institute of Technology (MIT) .

Common sense is a set of facts known by most people living in a particular culture, "covering a great part of everyday human experience, knowledge of spatial, physical, social and psychological aspects. In short, common sense is the knowledge shared by most people in a particular culture [1].

OMCS-Br project has been collected common sense of a general public through a web site. Common sense is then processed and stored in a knowledge base as a semantic network called ConceptNet where the nodes represents concepts and they are connected through arcs that represent relations according to the Marvin Minsky's knowledge model [9]. This base intends to reflect a basic knowledge structure near human cognitive structure. In the narrative game the common sense base can support the master in the characters' definition, and also at the story contextualization considering children's culture, promoting what can be called "just-in-time" context-aware sensible stories [1].

OMCS-Br project web site can be accessed by anyone through http://www.sensocomum.ufscar.br. After entering, the person can register and have access to various activities and themes available in this site. One of the themes available is about Children's Universe, which allows people to talk about situations, objects and characters existing in the Children's Universe, such as Folklore, Fairy

Tale, among others. Most of the activities and themes are templates as shown in Fig. 3. Template **assustar as crianças é uma característica do(a) personagem** saci pererê, (in English, **scare children is a characteristic of the character** saci pererê).

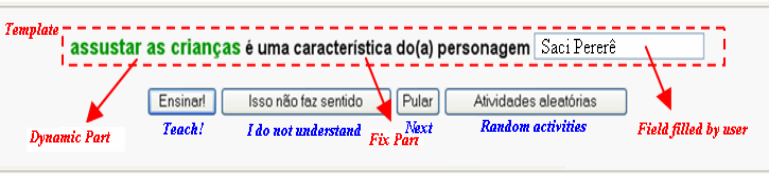


Fig. 3. Example of Children's Universe template

Templates are simple grammatical structures. They have fix and dynamic parts. Dynamic parts (green parts) change when they are presented to users. They are filling out with data from other users' contribution already registered on the site. Therefore this base uses the stored knowledge to collect new one. Templates also have a field to be filled by users considering their everyday experiences and knowledge. In short that represents for them a common sense fact. Words typed by users are stored. These words are in natural language. The phrases (template filled out by contributors) have to be processed by an engine that is based on Marvin Minsky´s theory about knowledge representation and then such engine generates the semantic network called Conceptnet, shown in Fig. 4 [14].

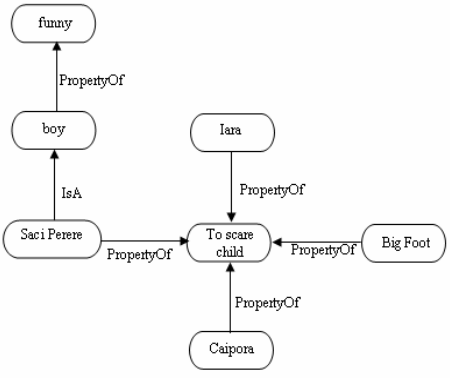


Fig. 4. Conceptnet example [14]

It was developed a group of templates composing a theme to collect common sense knowledge from people about the Children's Universe and use it in the narrative game. But this game uses all the common sense knowledge base, in short it uses all the information collected from other themes and activities.

Fig. 5 shows how the narrative game uses common sense knowledge to help teacher to define and to tell the stories. In this game, the common sense information is obtained through cards, which are presented on the master's interface. These cards allow him to use common sense knowledgebase in the story script and definition. For example, through this common sense knowledge (I) teacher can obtain characters or/and their characteristics.

The teacher can combine such information with the story that he wants to tell and to define the characters and their profiles and personalities (II). Each player needs to

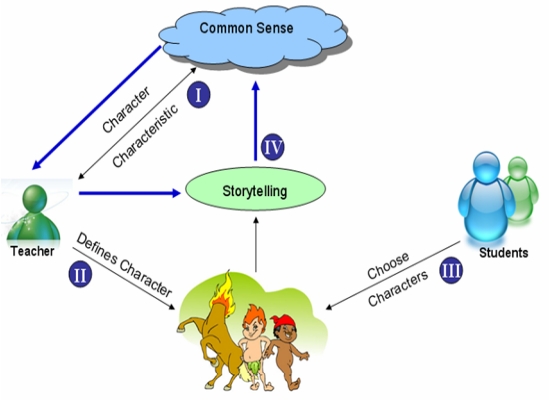


Fig. 5. Use of common sense in Contexteller

choose a character to participate in the story (III). During the story the teacher also has support from common sense (IV) to analyse the context and see how to interfere on the narrative, whenever she or he thinks it is necessary. As followed there is an explication about how the teacher uses this knowledge to create the game and to conduct the story.isplayed equations or formulas are centered and set on a separate line (with an extra line or halfline space above and below). Displayed expressions should be numbered for reference. The numbers should be consecutive within each section or within the contribution, with numbers enclosed in parentheses and set on the right margin.

Five steps are needed to create the game:

First, it is necessary to choose which common sense information the teacher wants to access, because of the approach that has to be sensitive to the culture. Stories should be to specific groups, considering their context, i.e., region, age, etc. According to that, common sense knowledge can be filtered and taking into consideration only the knowledge collected from the desired profile in order to contextualize the design for the target group.

Second, it is defined which students are going to participate. Then, defined the players, the teacher can access the whole story and how each student tell the story collaboratively. Therefore, the teacher could observe children's behaviour, evolution and growth, for example, comparing children's attitude in the first and last stories.

Third, it is necessary to define a subject and a title for the story. In this step the teacher uses common sense typing a word on the common sense card, and receiving contextualized information about this word. For example, the teacher types: "forest" and obtains characteristics, character, and other data that their students believe exist in the forest. Such as: Place with many three, animals, vegetation; there are bears, lobes, monkeys; and some other characters, for example, from the Brazilian folklore. This information can be used to define a title.

Step four is shown in Fig. 6. In this stage the teacher needs to define the six characters: one represents her/him and the others represent students. There are two common sense cards to define characters´ names and characteristics.

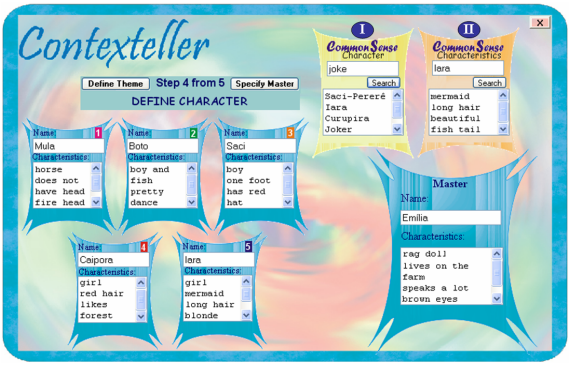


Fig. 6. Define Character

On the first card (I), the teacher types a characteristic and through searching the common sense knowledge base obtains the characters´ names. For instance, if the master wants to use a character in the story that likes to joke, to trick and to scare children, he/she can type these characteristics on the card. Through the common sense knowledge base can be seen the following characters: Saci-Pererê, Iara, Curupira, Caipora (from the Brazilian folklore), Joker (from Batman's), among others.

In the second card (II), it is possible to obtain the characters´ characteristics when characters´ names are written on the card. For example, some characteristics coming up from Iara´s character are: a mermaid, long hair, beautiful, a fish tail. The teacher can join such information to the story to define the characters, their profiles and personalities.

Fig. 7 illustrates the fifth and last step in which teacher needs to define the values for Magic and Force, and find an image to represent his characters.

After these steps, students choose a character to participate, in a similar interface shown in Fig. 6, define Magic and Force values, and find an image (Fig. 7). This feature allows the student to express himself not only through the story but also through the image. He can choose an image that makes his character sad, joyful, angry and so on. These emotions and their expressions can be a feature considered by the teacher when observing and developing the story.

During the stories the teacher can get support from common sense knowledge. Fig. 8 illustrates a situation where Iara´s character does not play because she is very worry about the fire in the forest. The master searches for fire in the common sense card and continues the story with the help of contextualized information.

Through this search the master knows that children in that community believe that Iara can be helped by Caipora because the latter is responsible for taking care of the forest and preventing fire. This character exists in the story, and then, the master talks to Caipora whether she can help Iara.

Searching on fire, Caipora shows up and if the master does not know who her and how she can help Iara, it is possible to search about Caipora to obtain some information about her, such as: prevent fire, care of the forest, etc.

There are many characters that prevent fire and take care of the forest but in a specify region children know that Caipora has these characteristics. If the teacher obtains this information through common sense, he/she can tell the story considering the



Fig. 7. Specify Character



Fig. 8. Master's interface

students´ reality. In order to provide common sense information about any Brazilian region, the teacher selects a filter in the initial phase. This filter is used to obtain the common sense from a certain community or group of people, for instance, teenagers from São Paulo in Brazil.

On the other hand, it is important to explain that the objective of the card is to help the teacher to find out what students know about a story or even about events, causes and consequences. Therefore, the teacher uses this information to tell the story, i.e., story definition and sequence. Because of that, players can feel connected to the characters, characteristics, scenarios and language of the story which the teacher defined using common sense knowledge help. Therefore, Contexteller does not teach common sense to the teacher but gives him/her a cultural feedback and help him/her to find out what the students´ knowledge is about the stories, facts and actions in order to tell the narrative according to the pedagogical goal proposed. It is necessary to clarify that there is no intention on teaching common sense, once it is a kind of knowledge acquired into the community, informally. Common sense was already proven to be a good way to contextualize the learning action considering the learners' culture and reality.

3 Conclusions

This paper has described an environment for online collaborative storytelling where the players jointly develop a story under the master's supervision. This game is meant to support a teacher to interact with students which have different social and cultural backgrounds.

Contextteller i.e., storyteller contextualized by Common Sense knowledge proposed here allows children to feel close and to identify themselves with the story. Therefore, players can express themselves through the character in their cultural context. They know and identify meanings through symbolism adopted by the teacher. These symbols can come from common sense knowledge of the children's community to define the character, the objects, in short, the story.

The teacher gets suggestions from the common sense to define the story and its sequence. This game also allows teachers to work with their students in a collaborative way. For example, the master defines a characters´ profile that represents their role in the work group. Therefore, during the story, he can consider this profile. For instance, the master asks a specific character a favour and he knows through the profile that character has some difficulties to perform the task. Therefore, he/she can observe how the character solves the problem, either asking help to others or solving it in a different way.

The master observes how this character is doing his work. If that character does it differently from his profile, the master asks other characters to think about that attitude. The master can also search on the common sense card what the character's abilities are.

In short, through common sense the master has contextualized information so that he/she can define and tell the stories with the students' participation. Because of that, students can learn to do anything cooperating each other.

Through Contexteller, students can also learn to express, to help and to be helped because they need to tell their stories, to help their friends to achieve an objective, and to known that they also need aid to achieve their objectives. Finally, master can observe how the student leads his character to interact with the others.

Acknowledgements. We thank CNPq, FAPESP and CAPES for partial financial support to this research. We also thank all the collaborators of the Open Mind Common Sense in Brazil Project who have been building the common sense knowledge base considered in this research.

References

1. Anacleto, J.C., Lieberman, H., Tsutsumi, M., Neris, V.P.A., Carvalho, A.F.P., Espinosa, J., Zem-Mascarenhas, S.: Can common sense uncover cultural differences in computer applications? In: Bramer, M. (ed.) Artificial intelligence in theory and practice - WCC 2006, vol. 217, pp. 1–10. Springer, Heidelberg (2006)
2. Bittencourt, R.J., Giraffa, L.M.M.: A utilização dos Role-Playing Games Digitais no Processo de Ensino-Aprendizagem. Technical Reports Series, Number 031, Setembro (2003)
3. Crawford, C.: The Art of Computer Game Design. Washington State University, p. 90 (1982)

4. Diaz-Aguado, M.J.D.: Educação Intercultural e Aprendizagem Cooperativa. Editora Porto, Porto (2003)
5. Fernandes, V.R.: What is RPG?. RPG - Dragon Magazine in Brazil (123) (2008)
6. Freire, P.R.N.: Pedagogia da autonomia: saberes necessários à prática educativa. 31 edn. Paz e Terra, Rio de Janeiro (1996)
7. Järvinen, A., Heliö, S., Mäyrä, F.: Communication and Community in Digital Entertainment Services. Prestudy Research Report (2002)
8. Lopes, L.M.C., Klimick, C., Casanova, M.A.: Relato de uma experiência de Sistema Híbrido no Ensino Fundamental: Projeto Aulativa. Revista Brasileira de Aprendizagem Aberta e a Distância, Janeiro, São Paulo (2003)
9. Minsky, M.: The society of mind. Simon & Schuster (1987)
10. Oaklander, V.: Windows to Our Children: A Gestalt Therapy Approach to Children and Adolescents, p. 335. Gestalt Journal Press (1988)
11. Papert, Seymour. Logo: computadores e educação. J A Valente (Trad.). Brasiliense, SP (1985)
12. Pastoreau, M.: Dicionário das cores do nosso tempo. Editorial Estampa, Lisboa, 188 p. (1997)
13. Piaget, J.: Judgement and Reasoning in the Child, 268 p. Littlefield Adams, Richmond (1999)
14. Silva, M.A.R., Anacleto, J.C.: A Narrative Game Culturally Contextualized by Common Sense Modeled as a Semantic Network. In: WSWEd@SBIE – Workshop on Semantic Web and Education (2008)
15. The Education Arcade. Revolution. Disponível em: Julho (2008), http://www.educationarcade.org/node/357
16. Tobaldini, M.A., Brancher, J.D.: Um RPG Educacional Computadorizado e Missões Contextualizadas com seus Ambientes. In: Anais do XV Seminário de Computação, pp. 85–96 (November 2006)
17. Vygotsky, L.: A formação social da mente. Martins Fontes, São Paulo (1987)

Applying the Discourse Theory to the Moderator's Interferences in Web Debates

Cristiano Maciel[1], Vinícius Carvalho Pereira[2], Licinio Roque[3], and Ana Cristina Bicharra Garcia[4]

[1] Instituto de Computação, Universidade Federal de Mato Grosso
Rua Fernando Correa, S/N Cuiabá, MT, Brazil
[2] Faculdade de Letras, Universidade Federal do Rio de Janeiro
Av. Horácio Macedo, 2151, Cidade Universitária, Rio de Janeiro, Brazil
[3] Departamento de Engenharia Informática, Universidade de Coimbra
Polo II - Pinhal de Marrocos, Coimbra, Portugal
[4] Instituto de Computação, Universidade Federal Fluminense
Rua Passos da Pátria, 156 sl 326 Niterói, Rio de Janeiro, Brazil
cmaciel@ufmt.br, vinicius.carpe@yahoo.fr, lir@dei.uc.pt
bicharra@ic.uff.br

Abstract. This paper presents a methodology for supporting the moderation phase in DCC (Democratic Citizenship Community), a virtual community for supporting e-democratic processes in e-life systems and applications. Based on the Government-Citizen Interactive Model, the DCC encompasses an innovative debate structure, as well as the moderator's participation based on Discourse Theory, specially concerning argumentative mistakes. Concerning the moderator's role, efforts have been made in order to improve the formalization of arguments and opinions while maintaining the usability of the platform. This research focuses on the moderator's participation via a case study and the experiment is analyzed in a Web debate.

Keywords: Web debates, Moderation, e-Discourse, e-Government, Decision-making, Virtual community.

1 Introduction

Electronic democracy (e-democracy) is a promising way to connect citizens and government, spurring discussions on collective matters or decision-making processes.

Many countries have been adopting different ways to promote citizens' involvement in decision making [1]. Referenda, plebiscites and popular initiatives are included in the different forms of direct demonstration of popular sovereignty prescribed in Federative Republic of Brazil's Constitution. Other governmental instances in different fields, such as collegiates in education, elect a smaller group amongst themselves, which decides on certain issues.

However, can our democratic process become electronic only by changing the ways of providing governmental services? E-democracy's traditional development has been following a relatively predictable model: at first, organizations offer information and add services; secondly, they initiate the attempt to add interactive tools.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 882–893, 2009.
© Springer-Verlag Berlin Heidelberg 2009

Implementing true e-democracy requires a careful and comprehensive methodology for constructing an effective infrastructure that is able to stimulate citizens to participate in decision-making.

Generally, applications with consultative and deliberative purposes are detected to be problematic [2]. One of these problems concerns the role ascribed to the moderator and, consequently, to his or her acting in environments projected with such intention. This paper presents a model for citizens to interact with the government and it discusses the moderator's role in acting in a debate component model. This proposal uses the concepts of Discourse Theory to enhance the quality of the debate. Furthermore, defining the moderator's interferences on discoursive bases provides an interdisciplinary alternative for a problem that is difficult to solve.

This research uses the Government-Citizen Interactive Model [2], which represents the different phases in a consultative and deliberative process. In general, the process begins with the government (the administrator) defining the type of popular manifestation and activity schedule. For socializing citizens, we propose the creation of a virtual community, structured according the type of manifestation, location and theme. This way, the model's components are integrated, such as debate, voting, socialization space, digital library and user's help. The debate phase, especially, requires a structure that permits discussing demands (topics to be debated on): registering opinions leads the citizen to justify his or her vote, indicating, furthermore, if his or her opinion goes against or for what is being discussed, or even if it is rather neutral. Justifications are thus classified and remain available for consultation. Eventually, citizens vote. The moderator's existence and actions are also modeled and are the main focus of this paper.

In relation to citizens' sociability on the web, Virtual Communities (VC's) are used to reinforce human interaction in order to construct knowledge. VC's have caused changes in society, modifying people's life in social aspects, in relation to technological innovation, as a communications medium and permitting the exchange of experiences. Nowadays, VC's have many social characters, not focusing on democracy and not stimulating citizens to participate in effective decision-making. Among other questions, this research investigates VC's modeling for e-democratic purposes and conduces a case study using the Democratic Citizenship Community (DCC) [3][4].

This paper is structured as it follows: after this introduction, in the section 2 there are the theoretical foundations of the Discourse Theory, and the role of the moderator in Web environments is discussed. In the section 3, the Government-Citizen Interactive Model is detailed, focusing on how the moderator can interfere in a debate according to certain categories of argumentative mistakes, proposed with basis on the Discourse Theory. The section 4 describes DCC briefly. Next, the methodology and the data analysis are presented in a case study, also evaluating the user satisfaction. Finally, the conclusions and references are enlisted.

2 Discourse Theory and the Role of the Moderator

Virtual environments on the web, especially those in which a certain group interacts with a view to exchange information, require the presence of a member who is responsible for the moderating activity. This way, the electronic discourse

(e-discourse) can achieve higher quality, mainly when it is a means for members to make decisions.

For the moderator to be effective, it is recommended that his or her acting should be supported by an interdisciplinary approach, which gathers web technologies and linguistic studies on discourse. Such perspective stems from the Linguistic Turn [5], an intellectual movement which brought together language studies at textual level (discourse studies) and other fields of knowledge. This way, the following reflections from Linguistics are applied to the study of the moderator's role in web environments.

In line with the discourse theory, following a logical perspective, there are two ways of making a mistake when expressing an idea: misthinking with correct data or thinking correctly with wrong data [6]. There is certainly a third way to make a mistake: misthinking with wrong data. The failure, therefore, can result from a formal error (misthinking with correct data) or from a subject error (thinking correctly with wrong data). According to Garcia [6], one must not confuse the error itself (the misconception) with the thought that produced it. For the author, investigating the causes of the mistake is not a task for Logic (this is to be done by psychology, natural sciences, or maybe metaphysics), which must only describe its forms. Beliefs, superstitions and taboos are errors: Logic does not discuss them, it only proves that the consequent misconceptions stemmed from misthoughts. Such argumentative flaws may obstruct the debate in web environments, thus making it necessary to offer support to the moderator in order to warrant intelligibility in e-discourse, an essential condition for a dialogical construction of meanings.

An e-discourse is a goal-oriented argumentative communication process that is fundamentally supported and documented by electronic media. It can be part of a larger attempt and is often planned and managed by a neutral moderator [7]. Moderators try to minimize process frictions, keep the discussion open, clear and fair, and aim at a high-quality, informed and consensus-oriented result. Through electronic communication, more people can be involved, contribute with more ideas and perspectives, and arrive at a sustainable solution. Through documentation, the process becomes more transparent, auditable, reproducible, and transferable to new situations.

E-moderators should be people with a great breadth of knowledge and troubleshooting skills. They should be neutral third persons whose goal is to manage the process successfully. By applying Information and Communications Technology, moderators can act electronically (e-moderation). With e-moderation, the e-discourse may benefit from a balanced input, a fair division of labor, better conflict resolution and a reduction of difficulties in terms of performance, socialization, domination and coordination.

According to Paulsen [8] and Salmon[9], guidelines for moderation include, but are not limited to: selecting, briefing and preparing participants; establishing norms and guidelines; helping the group create common, measurable goals; developing a cohesive, safe and open environment; creating an open, interactional network of people; objectively examining group processes; moderating participant interaction; interpreting non-verbal interaction (i.e. emoticons); opening/closing topics/threads; writing short summaries; issuing a final report.

Voss and Schafer [7] propose a framework to characterize environments for e-discourses aimed at online communities. In this framework, the e-moderator, discourse meters, discourse ontology, discourse flow, and discourse management are key

factors. Hafeez and Alghatas [10] apply Discourse Analysis to study interaction in a VC. According to these authors, when it concerns the virtual community, the interaction among members and the moderators' attitude can build a conversation structure. Moreover, they introduce the possibility of any member, even if not formally named a moderator to act as such, in view of strategies to make the debate clearer, more cohesive and more coherent.

Concerning the correlated works, there are researches on natural language processing in discussion forums, on ontology and semantic web usage and on moderation in virtual environments. Besides, this paper provides as a contribution the use of Discourse Theory as a strategy for e-moderation, enabling differentiated interventions in online debates.

3 Government-Citizen Interactive Model

This section comprises a description of the Government-Citizen Interactive Model [2], developed with a view to organizing consultative and deliberative processes with e-democratic purposes. The main characteristics of this model are the differentiated debate structure, the moderator's participation, the possibility of voting on the debated issues, and the formation of a virtual community for socializing its users.

According to the model, briefly described, citizen's participation in a community in a certain e-deliberative process is structured with respect to regions and themes. These citizens can submit their names as moderators. Enrolled citizens will be allowed to post demands that interest them, which will be discussed in the debate environment, according to a previously-arranged schedule. The debate is organized as it is proposed in Democratic Interaction Language – DemIL [2], which classifies opinions, with their respective justifications, in the categories "for", "against" and "neutral". Moderation activities in the debate are to be carried out by the citizens who proposed to do so. Due to the way it is structured, recovering information in the environment is easy, as well as analyzing data quantitatively and statistically. After this phase, members are stimulated to vote, in definite turns. When the voting period is over, results are deliberated. It is suggested that there should be a socializing space for users to get to know one another; also, in order to exchange information in other formats of digital files, there ought to be a digital library. In the model, the government, preferably, is supposed to manage the system.

It is very important to consider the participation model [1] that is to be adopted to adapt this model, once there are specificities when carrying out, for example, a popular consultation like a referendum or like a focus group. The model must also incorporate non-functional characteristics that are vital for a governmental application, such as usability, accessibility, security, and data privacy.

The components of the Model for a Virtual Community have clear functions, as presented in figure 1. Considering that this paper focuses on the debate phase, the following components will not be described in detail: registering the citizen's profile, using the information library, using the social space, using the help menu, and deliberative result. The main components for consulting and voting are:

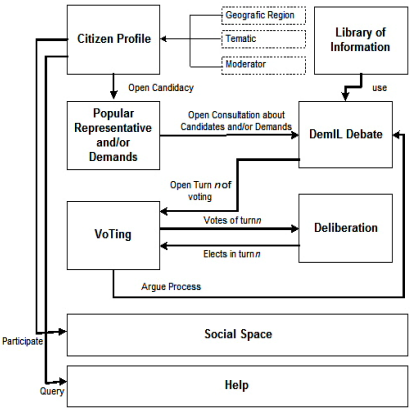


Fig. 1. Government-Citizen Interactive Model [2]

- Registering demands: the citizens who participate in the community must register their demands, that is, the issues they want to debate in the ongoing deliberative process. It is important to say that demands are also categorized into themes. Regarding the way a demand is written, there is one restriction: it must be put into words so as to permit the citizen to vote against, for or neutrally towards the demand.

- Participating in the DemIL Debate: through this component, citizens can exchange information, one of the primary characteristics of a democratic debate. In the DemIL Debate component, a forum structured with specific characteristics is modeled. In this forum, previously registered demands according to location/theme are discussed and foment opinions (arguments), which in turn can foment other opinions (counterargument); this characterizes a democratic exchange. An opinion has the following attributes:

- Author: citizen who posted the opinion;
- Date: record of the date of the posted opinion;
- Hour: record of the hour of the posted opinion;
- Type: an opinion can either be a justification to a demand, that is, an argument, or suffer the moderator's interference on the environmental theme (present theme).
- Justification: request for the citizen's position, in which he or she classifies the textually registered opinion "for", "against" or "neutral" in relation to the demand.
- Motivation: the moderator can interfere in the opinions posted in the debate. Citizens who have a more active political commitment usually act as moderators. The moderator interferes only when necessary and may explain his or her interference textually. The Discourse Theory [6] suggests that the moderator should employ certain types of interference, which will be described and exemplified in Section 3.1 in this paper. Another way to motivate participation in the debate would be moderators posing questions to users, but this strategy is not investigated in this paper. Opinion clustering is part of the proposed model, but it was not implemented in the developed community.

- Vote registration: a demand is put in discussion for people to vote for, against, or neutrally; there is a secret ballot and an anonymous justification. Votes are counted and the justifications registered in each demand are incorporated into the deliberative report.

3.1 Moderator's Interference

The moderator can interfere, reacting to comments posted in the community, so as to help clarifying utterances, stimulating the debate and warranting the good usage of the environment. The moderator only interferes if it is necessary and he or she can justify his interference textually.

Four types of interference to be employed by the moderator were proposed, based on the Discourse Theory [6], in relation to discourse mistakes: unclear opinion, inconsistent argumentation, excessive generalization and thematic deviation. In order to vouch for compliance to the terms of use, another type of interference was proposed: disrespect to the terms of use. For a better understanding, each type of interference is described and exemplified below. Example of a demand (topic) proposed for debate: Voting is a democratic exercise, so it should not be compulsory.

First type of interference - unclear opinion. At times, participants contradict themselves in their own discourses, supporting and denying the selfsame statement in a single post. Besides, misconstructed sentences can cause noise in communication, so that a member's manifestation may be unclear to others. E.g. «The compulsory nature of voting, so that some defend it and others condemn it.»

Second type of interference - inconsistent argumentation. Argumentation is based on two main elements: consistency of thought and evidence. When a member states his or her opinion but does not support it with relevant arguments, the development of the thought is inconsistent and the argumentation is flimsy [6]. E.g. «People should not be made to vote because it is not good.»

Third type of interference - Excessive generalization. One of the possibilities for structuring logical thought consists of induction, a method in which one takes particular premises to reach a general conclusion [6]. By means of generalizations, a participant tries to convince others of his or her opinion, but it may lead to untruths tainted with prejudice. E.g. «It is useless to vote, because all politicians are corrupt.»

Fourth type of interference - Thematic deviation. This is a common flaw involved in polemic debates, especially when passion deviates an individual from the debated issue, so that it replaces what was being discussed by some other, irrelevant claim. Thus, important facts become neglected, sometimes resorting to an emotional appeal. E.g. «Politics is present in every moment of our lives.»

Fifth type of interference - Disrespect to the terms of use. This type of interference is to be used when a community member disrespects the rules that are presented in the previously consented Terms of Use, when entering in an e-participation system. As instances of such disrespect, one may cite using bad language, posting illegal or pornographic material and violating others' intellectual property. E.g. "It irritates me to read the stupidities members post in this topic."

Studying these techniques to interfere on members' comments is innovative and it can be hard to be understood by moderators and users, especially if we consider the different educational levels of citizens. However, due to their scientific bases and to stimulating the discussion, they are applied in a virtual community for e-democratic processes. In the future, if the moderator's interferences prove to be effective for the debate, they will be studied in depth. Icons can even be projected to help represent graphically these types of interference.

4 Democratic Citizenship Community

Specially based on the Government-Citizen Interactive Model, the Democratic Citizenship Community (DCC) was specified, implemented and tested [3][4]. This section briefly presents the DCC. The DCC has interaction and communication resources, accessible by links in a tool bar, such as citizens' profiles, debate (demands register and discussion), voting, information library, social space and user's help. The system is accessed at the web address http://www.comunidadecdc.com.br/ (in Portuguese). See the DCC homepage in the Figure 3.

After registering or logging in to the DCC, the user is directed to his/her Profile. In the «Debate» link, after choosing the manifestation he or she is interested in, the user can «Register a new topic» to be discussed at the DCC. Also in the «Debate» box, previously registered demands are discussed. According to the DemIL language, in the «New debate» box the citizen must post his or her opinion or comment, choosing an option that defines his or her vote: for, against, or neutral in relation to the discussed demand. This post is added at the end of the list of the existing posts.

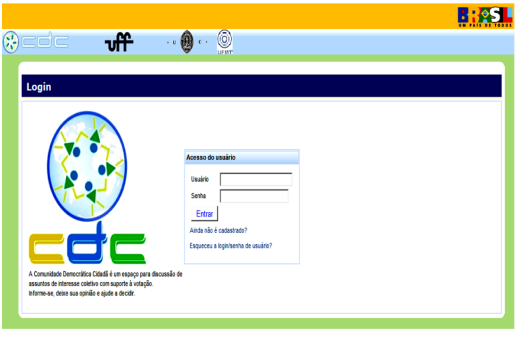


Fig. 2. DCC – Homepage

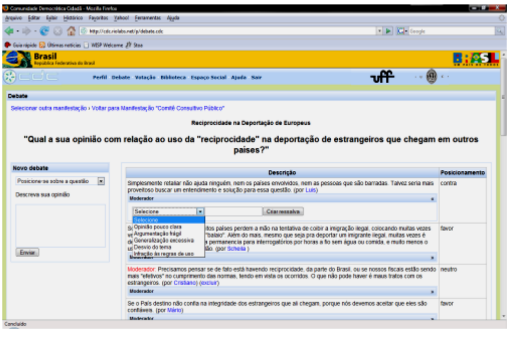


Fig. 3. DCC – Moderator

Both in «Voting» and «Debate», demands are listed and divided into themes and it is possible to vote for them in the previously scheduled period, which is established by the system administrator. When voting, the citizen can justify his or her vote, as well as visualize the justifications presented by other users.

For the citizen to obtain information, in order to be up-to-date when opining, there is a Digital Library with web links. In the socialization space of the DCC, where citizens visualize other members' profiles, one finds a news board. The DCC also has a help menu, structured by means of FAQs (Frequently Asked Questions), to clarify users' doubts about using the environment.

The system administrator visualizes the DCC differently, since he has distinct functions, as previously discussed. He has, in addition, the option «administration» in his tool bar.

5 Case Study

After implementing and, consequently, managing the DCC in a practical case, data were analyzed by means of usage statistics, with the aid of the analysis of logs registers in the administrator's view, Google Analytics tool and a survey made with the participants in the end of the process. The used methodology for the DCC implantation and the data analysis of the case study [11], focused on the moderators' participation, are presented in this section.

5.1 Methodology

Considering the Government-Citizen Interactive Model, a «Public Consultative Committee» manifestation was registered and a schedule was designed in the system establishing the phases of the consultative process. This schedule determines the deadlines for opening the debate, voting and finishing the activities. In this phase, the administration of the DCC registers interested locations and themes.

For the use of the DCC, the following phases were defined: 1) Registration of participants; 2) Registration of participants' demands; 3) Debate of opinions regarding the demands; 4) Voting; 5) User satisfaction survey; and 6) Deliberation of results. As suggested by users, the word «topic» replaced «demand» in the DCC's interfaces. However, «demand» is kept in the analyses.

During the initial contact between the user and the DCC, when enrolling, he or she registers his or her interest in acting as a moderator. After users register demands, moderators are allocated by the administrator according to themes. When enrolling, the user has access to the terms of use of the DCC, and it is necessary to agree with them to fulfill the enrollment.

The schedule for the «Public Consultative Committee» manifestation was designed to be completed in 20 days: 15 destined to debating and 5 to voting. The request to participate in the experiment was sent by e-mail, through the graduation and post-graduation lists of the institutions involved in the research. During the deliberative process, four warnings were sent to participants' mailboxes by the system, with the intention of stimulating participation in the discussions, explaining the moderators' role, reminding of the voting period, and demanding the fulfillment of the user satisfaction survey. Throughout the process, many e-mails from the users were answered, explaining general doubts about the application and expressing gratitude for compliments, criticisms and suggestions sent to the administration.

5.2 Data Analysis

This section analyzes the DCC users' demographic profile, their degree of interest in diverse aspects of debating and voting, as well as the satisfaction of use attested by the DCC members. Due to the small extent of the sample, data were not statistically presented.

Demographic Profile. The sample was composed of volunteer undergraduate and graduate students of the universities involved in the research: UFF and UC. It applied to a deliberative process, according to interests of the institutions. As the application presented the option «Invite a friend» in the social space, there were also external participations. Seventy-six individuals were interested in getting to know and taking part in the DCC, fulfilling the enrollment form. Among them, 67 were from Brazil and only 9 from Portugal. The participants' average age is 30 years old; 88% are students; the remaining are teachers. It is believed that this difference is due to the voluntary aspect of the participation, to individuals' particular interests on specific issues and to the difference between the countries' academic calendars. Another factor to be considered is the «social presence», since most of the Brazilians who were personally involved knew the research executor, collaborating with the experiment.

Registers in Debate and Moderation. Nine issues were suggested by 8 different users to be discussed in relation to the themes registered in the system (a member suggested 2 issues). There was greater interest on education, but there were also polemic topics, such as foreigners' deportation, which is currently being discussed by different media, and abortion, which has already been a referendum issue in Portugal. In general, demands were well-written by their authors, promoting the exchange of ideas and favoring the debaters' positions. Registered demands (with the respective acronyms) are presented in Table 1.

Table 1. Demands posted in the DCC

Demand's title	ID
Space of socialization for students	D1
Reciprocity in Europeans' deportation	D2
Academic publishing	D3
Support to access to higher education	D4
Is there life during postgraduation?	D5
Distance Education: a solution to democratize Brazilian education	D6
Compulsory voting	D7
Abortion – for or against?	D8
Good healthcare system should be for free	D9

Opinion posts for each of them and debaters' positions (for, against or neutral) are presented in Table 2.Even if a user had already posted an opinion and then decided to do it again, both posts were taken into consideration.

In relation to the debates, it is noteworthy that the visibility of others' opinions and positions stimulated the debate at times; sometimes, however, it inhibited users' opinion posting. Labeling positions as «for», «against» and «neutral» is positive, because it allows users to make a decision and ideas to be discussed, which is good for the debate.

Table 2. Posts per topic in the DCC

ID	Opinion posts	For	Against	Neutral
D1	22	21	-	1
D2	19	7	5	7
D3	9	4	2	3
D4	9	-	8	1
D5	8	5	2	1
D6	17	13	2	2
D7	14	7	5	2
D8	5	1	3	1
D9	2	2	-	-

Some comments can be articulated in relation to moderators' participation, although this role has been hardly tested so far. As a whole, 13 posts were moderated. Demands D1 and D2 had two posts each, in which moderators made interferences of the «Unclear opinion» type, but these had no reply from the post authors. In D3, there was an interference of the «Excessive generalization» type, to which the post author replied, trying to defend his or her idea. Two interferences condemning «Unclear opinion» were made in D4, but only one of them was answered by the author, who tried to clarify his previous utterance. Also in D4, two posts presented «Excessive generalization», one of which suffered interference from two moderators; in addition, two other posts were moderated because they presented «Disrespect to the terms of use». A «Thematic deviation» was found in a post in D6, to which the author replied. Two posts were moderated in D7: one because of a «Thematic deviation»; the other, because of an «Inconsistent argumentation», but their authors did not react to the interferences. It is also important to note that the interferences labeled «Thematic deviation» and «Disrespect to the terms of use» were punctuated negatively in the DMM (Degree of Maturity Method), since they were considered invalid justifications.

User Satisfaction Survey. By means of an online survey, thirty participants eventually filled in a user satisfaction survey, regarding their experience in the DCC. Participants evaluated the project and the use of the application for debating and voting. The administrator's participation was considered very good by 63.3% of the users and so was the moderators' by 50% of the users. The reaction to the moderators' participation was investigated in other questions of the user satisfaction evaluation tool, once a differentiated moderation was proposed for the debates. For 86.7% of the users, moderators' participation helped to stimulate the debate. On the other hand, 6.7% thought it was irrelevant and other 6.7% issued no opinion about it. Ninety percent agreed that the moderators' participation helped keep the environment orderly, but 6.6% disagreed and 3.7% did not know what to say about it. A worrying question, due to the fact that it is an innovative proposal, was the use of the moderation categories, which were satisfactory for 73.4% of the users, but dissatisfactory for 6.7%, whereas 20% had no opinion regarding the issue. A dissatisfied member justified his opinions as follows: «As far as I'm concerned, a moderator should only intervene when the discussion becomes offensive or improper, so as to morally damage a physical/legal person; otherwise, it might constrain an ethical expression» *(sic)*. Furthermore, another user stated that «The moderator can also intervene when someone intends to bias a discussion, in order to prevent opinions based on «populism»» *(sic)*. These

opinions show different expectations concerning the moderator's role and to his or her preparation to adequately act as such.

6 Conclusions

Integrating consultative and deliberative environments for popular participation in democratic issues and creating virtual communities make it possible to model decision-making processes. In relation to the Model presented and to the experiments that were conduced, some assumptions are possible.

If, in different methods of participation that require physical presence, it is hard to preside over and motivate the debate, in a virtual environment it is necessary to stimulate groups in a continuous and controlled manner, combining both interpersonal communication competences and technical and management competences. Thus, the role of the moderator is essential, and not identifying individuals with such capacities limits the application of the Model.

Concerning the types of interference that should be made by the moderator, it is possible to say that they aim at warranting the quality of the debate, although it may be hard for participants to understand them. So far, explanations and examples for each type of interference were added to the help menu. It is necessary to undergo further studies on the proposed textual categorization and on graphically representing (by means of icons) these types of interference.

Of its own accord, the presented model does not warrant the quality of the interaction between government and citizen. The interface designer can increase this quality if he or she considers aspects of computer-mediated communication important, such as usability and members' sociability, among others. Moreover, it is essential to consider the need for bases on other social theories, encompassing the possibility of empowerment ascribed to systems, everyone's right to accessing information (transparency) and limitations imposed by digital literacy. Thus, it is necessary to discuss criteria, defining evaluation parameters and tools that can guide the designer when developing applications in specific areas.

It is important to point out that technologies must be faced in e-democracy as means, not ends. They should not be regarded neutral, because they carry values, concepts, social views, conflictive, privileged and excluding processes, among others. Technologies, as DCC, were created to solve concrete problems, thus having political and social content. By themselves, they can't warrant citizens' active and critical participation in public interest issues. Success in a consultation and voting process is not directly related to the employed means, that is, technology, but to citizens' and government's motivation and interest on making it possible.

Finally, it should be noted that the DCC is used as the test application of the Maturity in Decision-Making method (MDM) [11], used for measuring, from a set of indicators, the participation of individuals in deliberative groups.

Future works are intended to evaluate whether the proposed categories for the moderator's interferences have been properly used. From a technical perspective, it is intended to study how moderation could be partially automatized, maybe using ontologies and semantic web for debate structuring, and how debates' content could encompass a recommendation system based on social network analysis.

References

1. Rowe, G., Frewer, L.: Public participation methods: a framework for evaluation. Science, Technology & Human Values 25, 3–29 (winter 2000)
2. Maciel, C., Garcia, A.C.B.: Modeling of a Democratic Citizenship Community to facilitate the consultative and deliberative process in the Web. In: Proceedings of the International Conference on Enterprise Information Systems (ICEIS 2007), Funchal, vol. 9, pp. 387–400. INSTICC Press, Portugal (2007)
3. Maciel, C., Garcia, A.C.B.: Design and Metrics of a Democratic Citizenship Community' in Support of Deliberative Decision-Making. In: Wimmer, M.A., Scholl, J., Grönlund, Å. (eds.) EGOV 2007. LNCS, vol. 4656, pp. 388–400. Springer, Heidelberg (2007)
4. Maciel, C., Roque, L., Garcia, A.C.B.: Democratic Citizenship Community: an e-Democratic application. In: Electronic Democracy: Achievements and Challenges, European Science Foundation - LiU Conference. Vadstena, Sweden (November 2007), http://www.docs.ifib.de/esfconference07/conf_programme.html
5. van Dijk, T.A.: O giro discursivo. In: Iñiguez, L. (ed.) Manual de análise do discurso em ciências sociais. Vozes, Rio de Janeiro (2004)
6. Garcia, O.M.: Comunicação em prosa moderna, 9th edn. Editora da Fundação Getúlio Vargas, Rio de Janeiro (1981)
7. Voss, A., Schafer, A.: Discourse Knowledge Management in Communities of Practice. In: DEXA 2003. IEEE Computer Society, Los Alamitos (2003)
8. Salmon, G.: Developing Learning Though Effective Online Moderation. Active Learning 9, 3–8 (1998)
9. Paulsen, M.F.: Moderating educational computer conferences. In: Berge, Z.L., Collins, M.P. (eds.) Computer mediated communication and the online classroom: Distance learning, Washington, APA, vol. III, pp. 81–89 (1995)
10. Hafeez, K., Alghatas, F.: Knowledge Management in a Virtual Community of Practice using Discourse Analysis. The Electronic Journal of Knowledge Management 5(1), 29–42 (2007)
11. Maciel, C.: Um método para mensurar o grau de Maturidade da Tomada de Decisão e-Democrática. Niterói (RJ), 2008, 230 p. PhD Thesis in Computer Science, UFF, Niterói, RJ (2008)

ExpertKanseiWeb: A Tool to Design Kansei Website

Anitawati Mohd Lokman[1], Nor Laila Md. Noor[1], and Mitsuo Nagamachi[2]

[1] Faculty of Information Technology and Quantitative Sciences, Universiti Teknologi MARA40450 Shah Alam, Malaysia
{anita,norlaila}@tmsk.uitm.edu.my
[2] International Kansei Design Institute, Japan

Abstract. In this paper we describe our research work involved in the development of a design tool for developing Kansei website. The design tool to facilitate Kansei web design is named, ExpertKanseiWeb and was developed based on results obtained from the application of the Kansei Engineering method to extract website visitors' Kansei responses. From the Partial Least Square (PLS) analysis performed, a guideline composed from the website design elements and the implied Kansei was established. This guideline becomes the basis for the systems structure of the design tool. ExpertKanseiWeb system consists of a Client Interface (CI), system controller and Kansei Web Database System (KWDS). Client can benefit from the tools as it offers easy knowledge interpretation of the guideline and presents examples to the design of Kansei website.

Keywords: Design tool, Emotional design, HCI, Kansei website, Partial Least Square analysis, Web design.

1 Introduction

The discipline of design science emphasizes the integration of cognitive, semantic and affective elements in the conception and development of designed products. Designers of IT artefacts have begun to address affective or emotional elements in their products and significant amount of work is seen in the design of mobile phones. However, the literature does not exhibit significant work on artefacts such as websites.

In this paper, we report the results of our research on affective website interface design. In our earlier work [1] [2] we put forward the conceptualisation of Kansei website to add up to the literature on design informatics by establishing a design methodology to embed website visitors' emotional impression in its interface design. Extending this work, we produced a guideline for Kansei website design which is derived from the Partial Least Square (PLS) analysis. This guideline is composed of detail design elements and visitors' Kansei and holds a tremendous amount of knowledge which may pose some difficulty when one has to read and interpret the knowledge. As a solution, we developed a design tool which we name the ExpertKanseiWeb, to convey the guideline to website designers in a systematic approach. This paper presents the development work on this design tool.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 894–905, 2009.
© Springer-Verlag Berlin Heidelberg 2009

2 Emotional Design of e-Commerce Websites

Over the past decades, the studies on user experience in website design were focused mainly on cognitive functionality and usability [3] [4] [5]. They include features such as active links, loading time, colour, typography, content organization, navigation and etc. covering features that may influence user experience with the website. In recent years, the concentration has shifted towards addressing emotional experience of website [6] [7] [3] [8]. This is due to the evolution of websites function from conveying information to the extent of providing persuasive engagement with visitors through the lively process of perception, judgment and action. Furthermore, emotional engagement has been found to influence decision making, perception, attention, performance, and cognition [9] [10] [11].

Aligned with these views, we argue that e-Commerce websites should induce desirable consumer experience and emotion that influences users' perception of the websites to extend the outreach potential of the online business. Hence, we need to consider the emergence of the dimension of desirability in e-commerce website design.

Desirability emerged from the realization of the need to have new measures of users' experience driven by emotional factors [12] [7]. Norman [13], an advocator of emotional design discussed the notion of emotional design through elements of visceral, behavioural and reflective factors. His views, parallels the view of Englelsted (1989, as cited in [14]), who discussed three temporal categories of emotions - affect, emotion, and sentiment. We argue that in terms of e-commerce website emotional design for desirability, visceral factors or affect that is the emotional state that results from a response to the external stimuli is more pertinent. Mahlke and Thüring [15] studied affect and emotion as important parts of the users' experience with interactive systems, aiming to consider emotional aspects in the interactive system design process. While admitting that emotion cannot be designed, they assert the importance of deriving a method for recognizing users' emotion from emotional evaluation procedures.

Despite the gained recognition, the subject of emotional appeal of websites or desirability is often neglected as designers tend to pay more attention to issues of usefulness and usability [16] due to the availability of established design methodology that addresses aspects of usefulness and usability. Design method that incorporates emotional design requirements is lacking. In addition, numerous studies conducted on emotional design tends to look at minimizing irrelevant emotions related to usability such as confusion, anger, anxiety and frustration [10]. Therefore, it is necessary to seek for a suitable design method to handle design requirements based on emotional signatures of websites. To seek for the method we turned to one established method of engineering product emotions, i.e. Kansei Engineering which will be briefly described in the next section.

3 Kansei Engineering

Kansei Engineering (KE) is a technology that combines Kansei and the engineering realms to assimilate human Kansei into product design with the target of producing products that consumer will enjoy and be satisfied with. The focus of KE is to identify the Kansei value of products that trigger and mediate emotional response. The KE process implements different techniques to link product emotions with product

properties. In the process, the chosen product domain is mapped from both a semantic and physical perspective. In terms of a design methodology, the approach of KE is to organize design requirements around the emotions that embody users' expectations and interaction [17]. KE has been successfully used to incorporate the emotional appeal in the product design ranging from physical consumer products to IT artifacts. Due to its success in making the connection between designers and consumers of products, KE is a well accepted industrial design method in Japan and Korea. In Europe, KE is gaining acceptance but is better known as emotional design.

4 Research Method

As seen in Fig 1 below, we divided the research into three phases. In the process of Kansei measurement, we adopted KE methodology to quantify website visitor's Kansei responses. Result from Phase I is then analysed statistically using Partial Least Square analysis to identify interrelations between design elements, influence of design elements to Kansei and link between Kansei and design elements. This lead to the establishment of guideline to the design of Kansei website, as anticipated in Phase II. Result of Phase II becomes the basis for the system structure of the design tool, the ExpertKanseiWeb. Details of research phases are described in Sections 5, 6 and 7.

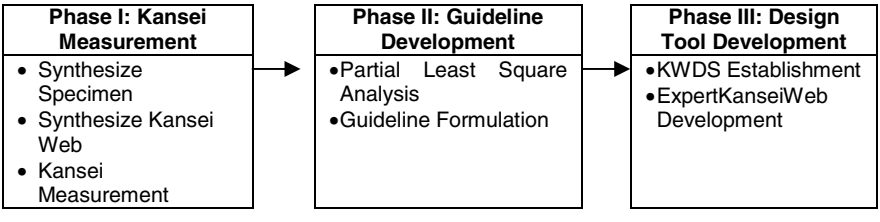


Fig. 1. Research method

5 Phase I: Kansei Measurement

Phase I begins with selection of specific domain. It is important to control the domain and subjects as different domain will induce different Kansei. Specific target market group must be used as experiments subject, so that the intended Kansei could be measured accurately. Failing which will lead to confusion during Kansei measurement and yield invalid result. The context of web application chosen for this work is the design of e-Clothing websites where emotional appeal is assumed to be significant. Correspondingly, the selected subjects are consumer with online shopping experience. Then, the study proceeds with synthesizing specimen, synthesizing Kansei Words, and Kansei Measurement.

5.1 Research Instruments

Initially, 163 online youth clothing websites were selected based on their visible design differences and were analysed following predefined rules on colours, design

elements, layouts, page orientations, and typography. From the analysis, 35 website specimens were finally used.

Kansei Words (KWs), which are used to represent emotional responses, were synthesized according to web design guidebook, experts and pertinent literatures. 40 KWs were selected according to their suitability to describe website. Among the synthesized words are 'adorable', 'professional' and 'impressive'. These KWs were used to developed checklist to rate websites, organized in a 5-point Semantic Differential (SD) scale.

5.2 Participants

120 undergraduate students from the Faculty of Information Technology and Quantitative Science, Faculty of Architecture, Building, Planning and Survey, Faculty of Business and Management and Faculty of Electrical Engineering from the researchers' university participated in the Kansei evaluation. From each faculty, exactly 30 students consisting of 15 males and 15 females were recruited. All of them have prior experience as web users.

5.3 Procedure

The participants were grouped according to their faculties. Four Kansei evaluation sessions were held separately for each group. During each session a briefing was given before the participants began their evaluation exercise. The 35 website specimens were shown one by one in a large white screen to all participants in a systematic and controlled manner. Participants were asked to rate their feelings into the checklist according to the given scale within 3 minutes for each specimen. They were given a break after the 15th website specimen, to refresh their minds. The order of checklist was also change to avoid bias. Each Kansei evaluation session took approximately 2 hours to complete.

6 Phase II: Guideline Development

Phase II begins with analysing results obtained from Phase I. There were three sets of data obtained from the study:

1. The dependant (objective) variables, y, i.e. the 40 sets of Kansei responses by 120 subjects.
2. The sample, i.e. the 35 website specimens.
3. The independent (explanatory) variables, x, the design elements (categories).

We calculated the average Kansei evaluation value of each samples obtain from all subjects from the experimental procedure. On the other hand, the initial investigation of design elements resulted 77 design items which composed from 249 categories. All three sets of data are used in performing the Partial Least Squares analysis used in the study to obtain the intended output. The three sets of data are the contributing component in the development of the guideline.

6.1 Partial Least Square (PLS) Analysis

PLS analysis was performed to discover relations between y (Kansei) and x (design elements). It is also used to identify influence of design elements in each Kansei, best fit and most unfit value for each design elements, and which sample induces what kind of Kansei. In the study, PLS analysis has been identified to be most suitable to handle the huge amount of x variables, and tens of y variables.

Table 1. PLS scores

	Kansei			
Category	**Adorable**	**Appealing**	**Beautiful**	**Boring**
BodyBgColor-White	-0.03655	-0.03699	-0.01674	0.024457
BodyBgColor-Black	0.006545	0.011992	-0.01374	-0.00265
BodyBgColor-DkBrown	0.060435	0.067045	0.018645	-0.03459
BodyBgColor-LtBrown	0.013248	0.011571	-0.00476	0.006006
BodyBgColor-Tone	0.013134	0.025984	0.028571	-0.03754
PageStyle-Frame	0.034036	0.025436	0.027154	-0.03955
PageStyle-Table	-0.04203	-0.03508	-0.02236	0.04195
DominantItem-Pict	0.046730	0.048044	0.030602	-0.04358
DominantItem-Adv.	-0.02968	-0.03225	-0.01741	0.019399
DominantItem-Text	-0.05612	-0.04549	-0.02663	0.050166
DominantItem-NotSpec	-0.01781	-0.02577	-0.02033	0.024059

We obtained coefficient values from PLS analysis. Table 1 shows a segment of the coefficient values. *Range* value is calculated to determine influence of each design category. *Range* is calculated using maximum and minimum value, where

$$Range = \mathrm{PLS}_{\mathrm{Max}} - |\mathrm{PLS}_{\mathrm{Min}}|$$

Mean of *Range* is calculated, where

$$\overline{Range} = \frac{1}{n}\sum_{i=1}^{n} Range_i$$

Each Kansei has means of *Range*, and if the mean value of a 'Category' is larger than $\overline{Range}$, the item is considered to have good influence in design. As a result, *Range* for every 'Category' having value bigger than $\overline{Range}$ implies the best fit 'Category' which highly influence consumer's Kansei in website design. To illustrate the result, shown in Table 2 is a segment of design influence for Kansei 'Adorable'.

Table 2. Design influence in Kansei Adorable

$\overline{Range} = 0.05$

KANSEI	ADORABLE		
Category	Range	Good Design	Bad Design
Page Color	0.114884	Brown	White
Product Display Style	0.106444	Filmstrip	Catalog
Header Menu Bg Color	0.106119	Grey	Blue
Left Menu Font Color	0.103703	White	Mix
Header Bg Color	0.102178	Grey	Blue
Face Expression	0.100237	Mix	None
Body Bg Color	0.100152	Dk Brown	White
Dominant Item	0.099800	Picture	Text
Header Font Size	0.096507	Not Text	Medium
Main Text Existence	0.088132	Not Exist	Exist
Main Bg Color	0.085869	Brown	Lt Blue
Main Font Style	0.085822	Italic	Normal
Main Font Size	0.083240	Medium	Large
Right Menu Link Style	0.078679	Picture	Text

Table 3. Website Kansei

Sample ID	Adorable	Appealing	Beautiful	Boring	Calm	Charming
1	3.23352	3.04163	3.14276	2.39434	2.77595	2.93343
2	2.72056	2.75152	2.94337	2.84746	2.70364	2.81997
3	3.68517	3.70649	3.30493	2.66730	3.52251	3.55746
4	2.52333	2.47425	2.92539	3.32273	2.83322	2.56347
5	3.06786	3.01651	2.98098	2.83406	2.67317	2.66387
6	2.34030	2.41328	2.58091	3.56973	2.66560	2.25456
7	3.16084	3.13096	3.25413	2.58095	2.77491	2.90715
8	3.49967	3.45554	3.35909	2.30261	3.32644	3.33457
9	2.65392	2.56151	2.84483	3.31682	2.74922	2.23772
10	3.31305	3.32962	3.00556	2.72732	3.49567	3.19025
11	2.96722	2.85847	2.96223	3.08820	2.88259	2.68545
12	3.26369	3.35897	3.43048	2.40299	3.16667	3.22916
13	2.87722	2.75896	2.95498	3.19583	2.87777	2.57862
14	*2.12599*	*2.18171*	2.69230	**3.72111**	2.68575	*1.95756*
15	3.32669	3.24533	3.54718	2.44541	3.02450	3.12673
16	**4.00896**	**4.02631**	**3.66163**	*1.97005*	3.56817	3.69670
17	3.88432	3.87891	3.43218	2.31150	**3.66133**	3.69804
18	3.57208	3.75637	3.21562	2.83233	3.49431	3.50153

: Best fit *: Most unfit*

The column 'Category' lists design category result that has influence in 'Adorable' design. The column 'Range' shows values higher than $\overline{Range}$, sorted in descending order, to show influence of design category from highest to lowest. Column 'Good Design' lists highest PLS score within a category, which implies best fit value to 'Adorable' website. Column 'Bad Design' lists lowest score within a category, which implies most unfit value to 'Adorable' website. PLS scores has also enabled the identification of Kansei in relation to each sample website.

Table 3, which shows Kansei sample score segment, represents largest value as the best fit Kansei and vice versa to describe a website. This result has lead to the discovery of sample's Kansei and enable the visualization of which sample highly implied what Kansei.

Results of all the analysis enable the paper to devise guideline to the design of Kansei website interface. The guideline is a composition of Kansei and value of each category that has influence to the design of Kansei website.

7 Phase III: Design Tool Development

Phase III is the process of developing the design tool, which uses findings from phase II as basis in its structure. The tool called ExpertKanseiWeb targets to facilitate clients in the process of developing Kansei website. ExpertKanseiWeb offers several options which client can choose from. The following sub-sections describe the tool development.

7.1 Kansei Web Database System (KWDS)

Firstly, results from PLS analysis are used to construct Kansei Web Database System (KWDS), shown in Fig 2, targeting to streamline the Kansei website interface design guideline. KWDS consists of Kansei Word Database (KWDB), Design Element Database (DEDB), LOGIC and Kansei Design Database (KDDB). KWDB stores all Kansei words, DEDB stores all identified design elements, and LOGIC handles interrelation of Kansei and design elements, influence of design element to Kansei, and the implied Kansei of particular web design.



Fig. 2. KWDS

KWDS deploys the expert system for Kansei website design. The system, ExpertKanseiWeb, allows client to input KW via selection from a list of existing Kansei words into the Client Interface (CI). The KW will be processed to identify the semantic taxonomy in reference to the KWDB. Inference Engine will then handle the associated design with the KW, extract the design elements with the detail attributes from DEDB and LOGIC, and send a design example to be displayed on CI.

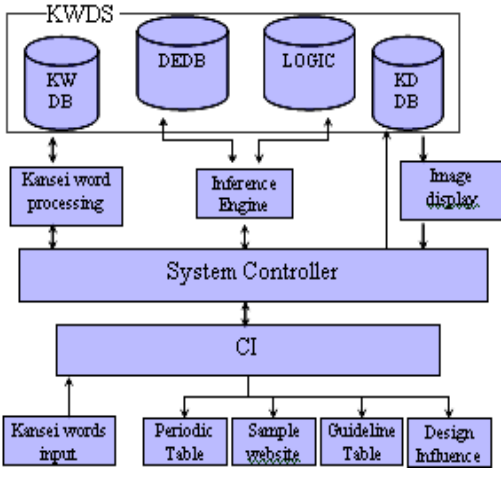


Fig. 3. System structure of ExpertKanseiWeb

The system structure of Expert Design Tool for Kansei Website is as illustrated in Fig 3.

7.2 The Client Interface (CI)

The presented CI as seen in Fig 4 is the main client interface to ExpertKanseiWeb. The interface offers selection of Kansei Word to client. Client can then select the type of display interface they desire. The first option offered is the information visualization in the form of periodic table (shown Fig 5), which is used to display the value of design elements, as a guide in designing certain Kansei website.

The periodic table is an arrangement of e-Commerce web design elements ordered by web page structure from top to bottom, left to right. Demonstrating the devised guideline in the form of periodic table is seen to be a solution to visualize the large amount of information as obtained from the study. The design elements could be represented in periodic form where the value is a repetition of patterns in terms of design elements. The value is shown when the intended Kansei Word is selected. Kansei Words can be selected from a drop down button on the CI, and the respective values will be displayed. In addition value for each design elements will be displayed when client hover a mouse over the elements.

The second option offers samples of websites that implies selected Kansei. A snapshot example of 'Feminine' Kansei website can be found in Fig 6. The left section of the page displays samples of website. The right section of the page displays several highest influences of design elements and its value, with option to see more influential elements. With this information, clients can have better clue in devising strategies to design a particular Kansei website. Selection of the intended Kansei also can be made from dropdown button provided at the top section of the CI, and display will change accordingly.



Fig. 4. Snapshot of the Main CI

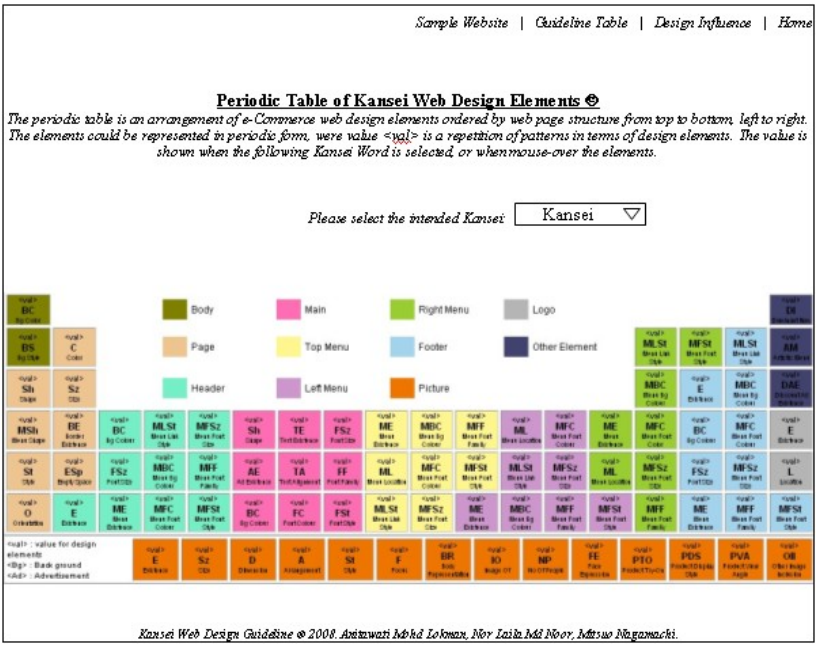


Fig. 5. Snapshot of Periodic Table of Kansei Web Design Elements

The third option provided in the Main CI (Fig 4), is to have a display of guideline in a table form. The table content will change when Kansei is selected from the drop down button. The table provides a full list of influential design elements and its value as a guide to design a particular Kansei website.

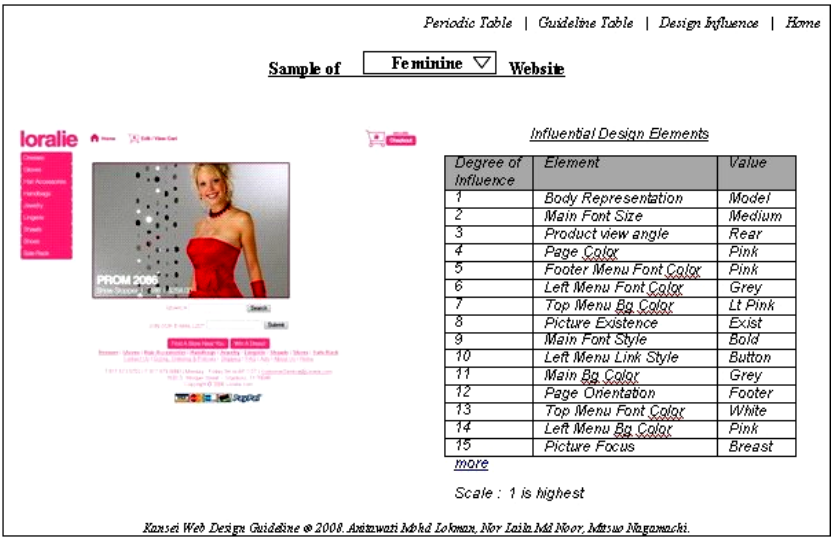


Fig. 6. Snapshot example of 'Feminine' Kansei website

Finally, the Main CI provides a fourth option, which is an interface display of design influence. The interface lays out influence of good design elements sorted in descending order to streamline elements from highest to lowest influence to the selected Kansei. The information is beneficial to client to figure out which element has high influence to certain Kansei website design. Options for Kansei are provided in drop-down button. A link to see a list of bad design influence is also offered, which is useful as an idea of what design value should be avoided when designing a particular Kansei website.

8 Conclusions

From our study, we have shown that it is possible to discover the emotional signature in website interface design. The PLS analysis has (1) revealed interrelations of design elements that contribute the influence towards the design of Kansei website and (2) linked between Kansei and design elements and website Kansei. Here, lies the biggest challenge of the study, i.e., in the process of translating Kansei responses to the underlying design elements. The heavy interaction between Kansei and design elements demanded careful attention. These results are used to formulate guideline which is a composition of design elements and the implied Kansei responses. The guideline holds a huge volume of information but one has to read and interpret the knowledge for it to be used. As a solution, the paper presented ExpertKanseiWeb, which lays down the interpretation offering easy access to the knowledge presenting idea to the design of Kansei website.

ExpertKanseiWeb help comprehend the large amount of data quick, consistent and accessible at any time. It offers an environment where the knowledge and the power of computers can be incorporated to overcome many of the limitations in human

capabilities. Additionally, the presented periodic table resolves the issue of visualizing large amount of data into one view.

ExpertKanseiWeb, streamline the guideline and delivers the knowledge of how design elements elicits Kansei. It facilitates client in devising strategies to improve website affective qualities, whereas positive affective qualities are proven to influence visitor's affective and eventually cognitive judgment [9] [14] [10] [15] [1]. Ultimately, the design of Kansei website will result in a paradigm shift from WYSIWYG (What You See Is What You Get) to WYSIWYD (What You See Is What You Desire).

Nonetheless, the study was performed focusing on e-Clothing and young consumers as target market group. Thus, the result may not produce globally applicable features. Additionally, although ExpertKanseiWeb is seen to provide solution to designing Kansei website, the effectiveness of the tool has not been tested. We will address these issues in our future work.

Acknowledgements. The research is supported by grants from the Ministry of Higher Education, Malaysia, under the FRGS grant scheme [Project Code: 5/3/2094].

References

1. Anitawati, M.L., Nor Laila, M.N., Nagamachi, M.: Kansei Engineering: A Study on Perception of Online Clothing Websites. In: The 10th International Conference on Quality Management and Operation Development (QMOD 2007). Linköping University Electronic Press, Sweden (2007)
2. Nor Laila, M.N., Anitawati, M.L., Nagamachi, M.: Applying Kansei Engineering to Determine Emotional Signature of Online Clothing Websites. In: Proceedings of the 10th International Conference on Enterprise Information Systems (ICEIS 2008), vol. HCI (5), pp. 142–147 (2008)
3. Li, N., Zhang, P.: Consumer Online Shopping Behavior. In: Fjermestad, J., Romano, N. (eds.) Customer Relationship Management. Series of Advances in Management Information Systems, Zwass, V (editor-in-chief). M.E. Sharpe Publisher
4. Nielsen, J.: Designing Web Usability: The Practice of Simplicity. New Riders Press (2000)
5. Lederer, A.L., Maupin, D.J., Sena, M.P., Zhuang, Y.: The Role of Ease of Use, Usefulness and Attitude in the Prediction of World Wide Web Usage. In: Robbins, S.P. (ed.) Organizational Behavior, 8th edn., p. 168. Prentice Hall, Upper Saddle River (1998)
6. Kim, J., Lee, J., Choe, D.: Designing Emotionally Evocative Homepages: An Empirical Study of the Quantitative Relations Between Design Factors and Emotional Dimensions. International Journal of Human-Computer Studies 59(6), 899–940 (2003)
7. Dillon, A.: Beyond Usability: Process, Outcome and Affect in human computer interactions. Paper presented as the Lazerow Lecture 2001, University of Toronto (March 2001)
8. Zhang, P., von Dran, G., Small, R., Barcellos, S.: Web Sites that Satisfy Users: A Theoretic Framework for Web User Interface Design and Evaluation. In: Proceedings of the International Conference on Systems Science (HICSS 32), Hawaii, January 5-8 (1999)
9. Tractinsky, N., Katz, A.S., Ikar, D.: What is Beautiful is Usable. Interacting with Computers 13, 127–145 (2000)
10. Norman, D.A.: Emotional Design: Attractive Things Work Better in Interactions: New Visions of Human-Computer Interaction IX, pp. 36–42 (2002)

11. Russell, J.A.: Core Affect and the Psychological Construction of Emotion. Psychological Review 110(1), 145–172 (2003)
12. Spillers, F.: Emotion as a Cognitive Artifact and the Design Implications for Products that are Perceived As Pleasurable, Experience Dynamics (2004)
13. Norman, D.A.: Emotional Design: Why We Love (or Hate) Everyday Things. Basic Books, New York (2004)
14. Aboulafia, A., Bannon, L.J.: Understanding affect in design: an outline conceptual framework. Theoretical Issues in Ergonomics Science 5(1), 4–15 (2004)
15. Mahlke, S., Thüring, M.: Studying Antecedents of Emotional Experiences in Interactive Contexts. In: Proceedings of CHI 2007, pp. 915–918. ACM Press, New York (2007)
16. Buchanan, R.: Good Design in the Digital Age. AIGA Journal of Design for the Network Economy 1(1), 1–5 (2000)
17. Nagamachi, M.: The Story of Kansei Engineering (in Japanese), Japanese Standards Association, Tokyo, 6 (2003)

Evaluation of Information Systems Supporting Asset Lifecycle Management

Abrar Haider

School of Computer and Information Science, University of South Australia
Mawson Lakes, South Australia 5095, Australia
abrar.haider@unisa.edu.au

Abstract. Performance evaluation is a subjective activity that cannot be detached from the human understanding, social context, and cultural environment, within which it takes place. Apart from these, information systems evaluation faces certain conceptual and operational challenges that further complicate the process of performance evaluation. This paper deals with the issue of performance evaluation of information system utilised for engineering asset lifecycle. The paper highlights that these information systems not only have to enable asset management strategy, but also are required to inform the same for better lifecycle management of the critical asset equipment utilised in production or service environments. Evaluation of these systems, thus, calls for ascertaining both hard as well as soft benefits to the organisation and their contribution to organizational development. This, however, requires that evaluation exercise identifies alternatives and choices and in doing so becomes a strategic advisory mechanism that supports information systems planning, development, and management processes. This paper proposes a comprehensive evaluation methodology for evaluation of information systems utilised in managing engineering assets. This methodology is learning centric, provides feedback that facilitates actionable organizational learning, and thus allows the organisation to engage in generative learning based continuous improvement.

Keywords: Information systems, Performance evaluation, Asset management.

1 Introduction

Since early 1990s there has been an increased research activity in development of performance measurement systems aimed at various organisational levels, covering a multitude of dimensions. This increase has been fuelled by certain business development theories that promote the need for employing performance evaluation as means for performance improvement, such as constraints theory, lean enterprise, and six sigma. The activity thus generated has resulted in development of numerous models, frameworks, techniques, and methods applied in the industry with varying levels of acceptance and success. However, the discussion on the effectiveness of performance evaluation has been centred on three views. The first view suggests that businesses do well if there are integrated and well structured performance evaluation methods in place that inform and provide management with improvement indicators [1]. In

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 906–917, 2009.
© Springer-Verlag Berlin Heidelberg 2009

contrast, there are researchers who have questioned the role of performance evaluation in general and individual performance evaluation methods in particular. For example researchers (See for example, 2, 3) suggest that employing console style performance evaluation methods such as balanced scorecard, makes little or no contribution to business performance improvement. There are, however, other researchers who suggest that performance evaluation is a business management activity and its success is highly dependent upon the approach used to implement it [4].

Performance evaluation of technology in general and Information systems (IS) in particular is complex issue, due to the conceptual and operational issues involved. Although, IS evaluation can be carried out at various stages in IS lifecycle, however, the most common evaluations are ex ante, ex post, and during operation. Depending upon the type of evaluation and the physical and organisational context of IS application, IS evaluation has different aims and objectives. This paper address the fundamental issue of 'how should IS utilised for engineering asset lifecycle be evaluated' and thus presents a comprehensive evaluation methodology for IS utilised in managing engineering asset lifecycle. It starts with a discussion of assets management and the role of IS in managing assets, followed by the issues involved in IS evaluation in general and IS for asset management in particular. The paper then presents a comprehensive IS evaluation methodology that accounts for the operational and conceptual issues, and exposes technical, organisational, social, and strategic dimensions of IS utilised for managing engineering assets.

2 Asset Management

The term asset in engineering organisations is defined as the physical component of a manufacturing, production or service facility, which has value, enables services to be provided, and has an economic life greater than twelve months [5], such as manufacturing plants, roads, bridges, railway carriages, aircrafts, water pumps, and oil and gas rigs. Oxford Advanced Learner's Dictionary describes an asset as valuable or useful quality, skill or person; or something of value that could be used or sold to pay of debts [6]. These two definitions imply that an asset could be described as an entity that has value, creates and maintains that value through its use, and has the ability to add value through its future use. This means that the value it provides is both tangible and intangible in nature. A physical asset should be taken as an economic entity that provides quantifiable economic benefits, and has a value profile (both tangible and intangible) depending upon the value statement that its stakeholders attach to it during each stage of its lifecycle [7]. Management of assets, therefore, entails preserving the value function of the asset during its lifecycle along with economic benefits. Consequently, asset management processes are geared at gaining and sustaining value from design, procurement and installation through operation, maintenance and retirement of an asset, i.e. through its lifecycle. Asset management is a strategic and integrated set of processes to gain greatest lifetime effectiveness, utilisation and return from physical assets [8]. The core objective of asset management processes is to preserve the operating condition of an asset to near original condition. IS are an integral part of an asset lifecycle management and perform various tasks at each stage of the lifecycle through data acquisition, processing and manipulation operations. However, the scope

of IS in asset management extends well beyond the usual data processing and reaches out to business value chain integration, enhancing competitiveness, and transformation of patterns of business relationships [9].

3 IS for Asset Management

The Institute of Public Works Engineering Australia [5] specifies minimum criteria to measure performance of IS for asset lifecycle management for contributions and compliance in terms of, justifications of planned levels of service; monitoring, and reporting and requirements; planned techniques and methodologies to enable cost effective asset lifecycle treatment options, such as risk management, predictive modelling, and optimised decision support; identification of task priorities and resources requirements; justification of the roles and responsibilities for various organisation units in relation to asset management activities; information requirements of asset lifecycle; and continuous improvement of asset management plan. Asset managing organisation have twofold interest in IS, first that they should provide a broad base of consistent logically organised information concerning asset management processes; and, second the availability of real time updated asset related information available to asset lifecycle stakeholders [10]. However, engineering organisations traditionally conform to technological determinism, where technology is viewed as the prime enabler of change and, therefore, is the fundamental condition that is essential to shape the structure and pattern of an asset management regime. Most engineering enterprises mature technologically along the continuum of standalone technologies to integrated systems, and in so doing aim to achieve the maturity of processes enabled by these technologies, and the skills associated with their operation [11]. Konradt *et al.* [12] further assert that engineering enterprises adopt a traditional technology-centred approach to asset management, where technical aspects command most resources and are considered first in the planning and design stage. Skills, process maturity, and other organisational factors are only considered relatively late in the process, and sometimes only after the systems are operational. However, human, organisational, and social factors have a direct relationship with IS [13, 14], which underscore the conceptual and operational constraints posed to effective IS implementation. It is, therefore, important to assess the performance of IS investments for compliance to their intended purpose and the contributions that they make in managing the asset lifecycle. This performance evaluation may be aimed at different dimensions of asset lifecycle management, such as, effectiveness, reliability, and cost effectiveness of design, operation, and maintenance.

4 Issues with Evaluation of IS for Asset Management

IS evaluation is often difficult and a wicked problem [15], due mainly to its varying roles in different organisations. Evaluation by nature is a subjective term and is defined in the Oxford Advanced Learner's Dictionary as, the process of judging or forming an idea of the amount, value, or worth of an entity [6]. Neely *et al.* [16] suggest that performance is the measure of efficiency and effectiveness of action; and

performance evaluation is the process of measuring accomplishments, where measurement deals with quantification of action and accomplishment illustrates performance. Tangen [17] takes the argument further and contends that performance evaluation represents the set of metrics used to quantify the efficiency and effectiveness of organisational actions taken towards achieving its objectives. The efficiency and effectiveness constitute the value profile that the organisational stakeholders attach to action in an organisation. In light of this discussion IS evaluation could be defined as "an assessment of value profile of IS to asset lifecycle using appropriate measures, at a specific stage of IS lifecycle within each stage of an asset lifecycle, towards continuous improvement aimed at achieving the overall organisational objectives".

4.1 Conceptual Limitations of IS Evaluation

Evaluation, conceptually, is a subjective activity that is biased and cannot be detached from the human understanding, social context, and cultural environment, within which it takes place. Evaluation, therefore, is influenced by the actors who carry out this exercise; and the principles and assumptions that they employ to execute evaluation. Scope of asset management spans engineering as well as general business or administrative activities. In addition, most of these activities are cross functional and even cross enterprise. For example, maintenance processes influence areas such as, quality of operations; safe workplace and environment; manufacturing management, and accounting. The outputs from maintenance are further used to predict asset remnant lifecycle considerations, asset redesign/rehabilitation, and planning for the support resources management. A single information snapshot is open to interpretation from different perspectives for various dimensions of quality and efficiency. Considering the fact that human interpretation shapes and reshapes over a period of time, the nature of evaluation also changes from time to time. Evaluation, thus, represents the existing meanings and interests that individuals or communities associate with the use of technology within the socio technical environment of an organisation. The focal point of socio technical perspective is the interactive association between people, IS and the social context of the organisation [18]. However, action is an important element of this interaction. This notion of action is contained in the structuration theory, which describes that it is facilitated and influenced by the social structure. People's interaction is, therefore, fashioned by the social structure and their actions persistently shape or transform social structure [19]. There is a dynamic relationship between technology, and the context within which it is employed and the organisational actors who interact with technology. This duality of technology is characterised by Orlikowski [20], who argues that technology is socially and physically constructed by human action.

When technology is physically adopted and socially composed, there is generally a consensus or accepted reality about what the technology is supposed to accomplish and how it is to be utilized. This temporary interpretation of technology is institutionalised and becomes associated with the actors that constructed technology and gave it its current significance, until it is questioned again for reinterpretation. This requirement of reinterpretation may grow owing to changes in the context, or the learning that may render the current interpretation obsolete. Technology, therefore, is not

an objective entity, such that it could either be evaluated without considering its interaction with social and human factors, or it could be evaluated in basic and one-dimensional economic terms. When IS evaluation is employed it is expected that it will expose a number of different dimensions of IS implantation, such as, financial, technical, behavioural, social, and management aspects of IS. Furthermore, these endeavours may be aimed at stakeholder satisfaction, role of IS, and IS lifecycle. These expectations change during the lifecycle of an IS. An ex ante or pre implementation is aimed at ascertaining cause and effect of technology; whereas, ex post or post implementation evaluation may be aimed at evaluation of strategic translation as well strategic advisory role of IS. Each of these dimensions, their related objectives and aims have their own theories, postulates, and evaluation criteria, which makes IS evaluation complicated and difficult.

4.2 Operational Limitations of IS Evaluation

Role of IT investments is no more considered as inwardly looking systems aimed at operational efficiency through process automation; in fact, it extends beyond the organisational boundaries and also addresses areas such as business relationships with external stakeholders, to deliver business outcomes. This complicates the process of decision making for IT investments, since this decision needs to take care of the impact of the investment on business processes and resources, as well as integration of these technologies with other systems. However, IS evaluation, generally has a narrow focus and involves people who cannot evaluate IT on anything other than technological dimensions [21]. Consequently, simplistic measures are adopted to measure the effectiveness of IS, while these efficacy criteria are aimed at process efficiency rather than its prospectus of organisational transformation. The measurement attributes involved in such IT investments, require both aspects of IT benefit to be taken care of i.e. soft benefits, such as stakeholder satisfaction, and customer relationship management; and hard benefits, such as cost, IS throughput. However, evaluation methods wanting in completeness render the accuracy and credibility of evaluation mechanisms questionable, in terms of their role as instruments of decision support. In IS evaluation the generally applied generic performance measures are financial measures, such as costs of implementation; technical measures, such as response time; system usefulness attributes, such as user satisfaction; and quality of the information [22]. IS, however, are social systems embedded within the organisational context and choosing criteria that encompasses evaluation of all the IS benefits is a difficult task. Teubner [23] points out these difficulties are due to a range of factors, such as,

a. Technical Embedding. Individual IS components are often embedded in the overall technological infrastructure, which makes it difficult to assess the performance of these individual components. For example, while evaluating the effectiveness of a condition monitoring system, it is difficult to quantify the contribution of individual sensors.
b. *Organisational Embedding.* IS infrastructure is an integral part of an organisation, and influences and is influenced by a number of organisational factors, such as culture and structure of the organisation. Consequently it has

progressively become difficult to take the impact of IS apart from these organisational aspects.

c. *Social Construction.* The social impact of IS is well documented, which makes it much more than just a technical solution. Impact of changes that IS implementation brings affect work practices as well as the intellect and working habits of employees. However, impact of IS on staff, social life of the organisation, and collective sense making, is intangible and is difficult to measure.
d. *Social Adoption.* IS adoption is a social process, since their use evolves over time and depends heavily upon skills of employees and culture of the organisation. It also means that IS may not start delivering desired results straight after their implementation. Evaluation criteria, therefore, needs to account for the time frame of IS lifecycle within which evaluation is to be carried out.

In light of above discussion, evaluation of IS for asset lifecycle management need to be comprehensive, which evaluates various hard and soft dimensions of IS and their impact on the organisation and strategic orientation of asset management; fit of IS with the information requirements of the asset lifecycle management processes; and contribution of IS in creating a unified view of asset lifecycle. This evaluation, thus, needs to provide insights into the effectiveness of asset lifecycle management through IS utilisation, and enable feedback on the relevance and fit of existing asset management strategies so as to enable continuous improvement.

5 IS for Asset Management Evaluation

IS for asset management evaluation depends upon three dimensions, i.e. the asset lifecycle processes that the IS enable; the elements of an IS, such as software, hardware, information, and skills; and the value profile attached to IS at each stage of asset lifecycle, such as efficiency, effectiveness, availability, compliance, and reliability of an asset solution. In order to have a complete measurement of the effectiveness of IS, different dimensions of IS must be assessed in terms of translating asset lifecycle management strategy into action, as well as advising strategy through decision support. However, in order to institutionalise a competitive IS based asset management regime, it is essential to focus on continuous improvement of asset lifecycle management processes rather than just fixing faults and errors. IS should enable constructive action oriented feedback, which enables continuous improvement in asset lifecycle management processes and the IS infrastructure that supports these processes. Such learning necessitates systemic thinking, shared vision, personal mastery, collective learning, and creative tension between the existing situation and vision. Having a generative learning focused performance evaluation methodology not only provides for the assessment of the tangible and intangible contributions of IS to asset lifecycle management, but also provides assessment of the maturity of IS infrastructure. Figure 1 illustrates an IS based asset management performance evaluation framework. It is a learning centric framework and accounts for the core IS based asset management processes as well as the allied areas where IS also make contributions. It therefore accounts for the soft as well as the hard benefits gained from IS utilisation in an asset lifecycle. This framework divides the asset lifecycle into 7 perspectives, where each perspective consists of processes that contribute to asset lifecycle

management. The framework begins with assessing the usefulness and maturity of IS in mapping the organisation's competitive priorities into asset design and reliability support infrastructure. The framework thus assesses the contribution and maturity of IS through four further perspectives before informing the competitive priorities of the asset managing organisation. In so doing, the framework evaluates the role of IS as strategic translators as well as strategic enablers of asset lifecycle management and enables generative learning. It means that instead of just providing a gap analysis of the desired versus actual state of IS maturity and contribution, it also assesses the information requirements at each perspective and thus enables continuous improvement through action oriented evaluation learnings.

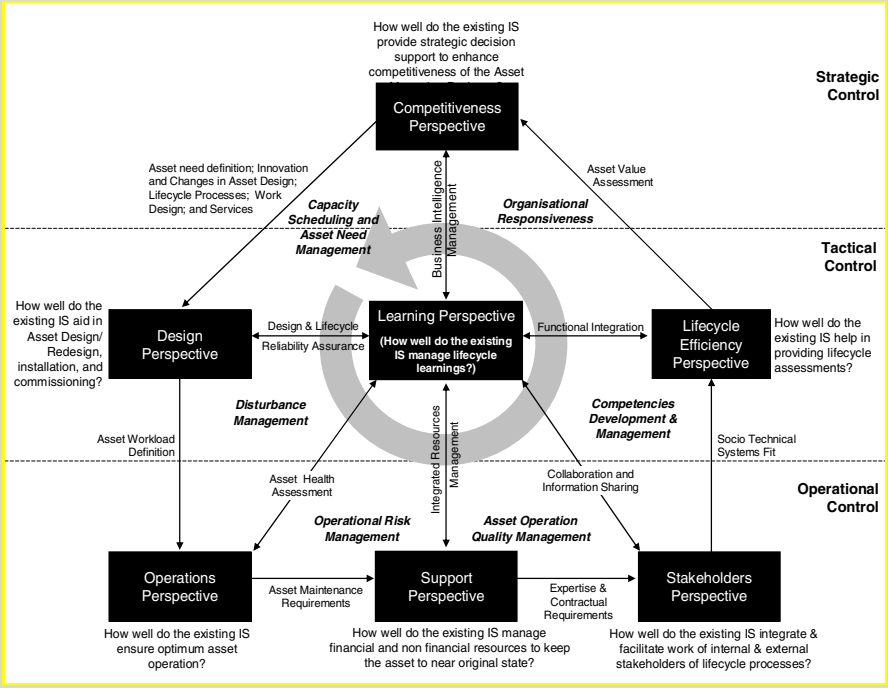


Fig. 1. IS Based Asset Management Performance Evaluation Framework.

The following sections elaborate on these points and uncovers the details of the framework.

a. *Capacity and Demand Management*. In a usual asset lifecycle asset demand and capacity specifies the nature of assets, as well as the types of supportability infrastructure required to ensure asset reliability through its lifecycle. The success of IS at this stage depends upon the availability, speed, depth, and quality of information regarding competitive environment of the organisation. This information allows asset managers to measure the demands of asset customers, which specifies the types of assets or the improvements required in existing asset configuration to address the customers' demands. The value profile that asset managers attach to IS at this point is of business intelligence management, so as to

aid the design of the asset as well as the support infrastructure. Within design perspective itself, there are a variety of information demands that the IS are required to fulfil. In a nutshell, the value profile of IS demanded by the asset designers specifies how the IS aid in asset design/re-design, installation, and commissioning. Nevertheless, each of these processes further consist of a series of activities that require an assortment of information to enable evaluations and alternative solutions, such that the organisation is able to chooses the best possible solution to asset design/redesign. These alternatives are arrived at after having considered a series of analysis that encompass the capability potential and associated costs for ensuring reliability of the asset operation. The success factor of IS in ensuring asset supportability and design reliability is the depth and coverage of supportability analysis, which provide a roadmap for the later stages of the asset lifecycle. These analysis not only specify the costs associated with supporting the asset lifecycle, but also identify other critical aspects such as the throughput of the asset, spares requirements, and training requirements.

b. *Disturbance Management*. Asset workload is defined according to its 'as designed' capabilities and capacity. However, during its operational life every asset generates some maintenance demands. During the asset operation stage, the critical feature of IS is to aid asset managers in managing disturbances. This requires availability of design as well as supportability information, as well as current information on the condition of an asset. Different organisations deploy different condition or health monitoring systems, such as sensors, manual inspections, and paper based systems. Nevertheless, IS at this stage need to be able to provide consolidated health advisories by capturing and integrating this information, analysing asset workload information, health information, and design information to enable speedy malfunction alarms and communication of failure condition information to maintenance function. It should be noted that many of the design errors surface during asset operation. It is, therefore, also important to assess if the existing IS report back these errors to the asset design function so as to ensure asset design reliability.
c. *Operational Risk Management*. The notion of risk signifies the 'vulnerabilities' that asset operation is exposed to, due to operating in a particular physical setting or specific work conditions. Nevertheless, the success of risk management is dependent upon factors such as availability of expertise to carry out maintenance treatments, availability of spares, maintenance expertise, maintenance project management as well as complete information on the health status and pervious maintenance history of the asset. The role of IS therefore need to be assessed for their ability to provide control of decentralised tasks and to ensure the availability of resources to keep the assets in near original state.
d. *Asset Operation Quality Management*. The aim of asset managing processes is to keep the asset to or near its original or as designed state throughout its operational life. Therefore once a disturbance has been identified, it becomes crucial to curtail its impact to minimum and to take appropriate follow up actions. These follow up actions not only involve the direct actions taken on the asset such as maintenance execution, but also involve sourcing of maintenance, rehabilitation, and renewal materials and expertise as well as the contractual agreements. At the same time with the growing attention being given to the

environment, it is equally important to ensure that the asset operation conforms to the governmental and industrial regulations, and to control the impact of disturbance on the environment. IS at this stage have a versatile role, and aid in maintenance and rehabilitation execution, enabling collaboration and communication, managing resources, as well as facilitating business relationships with external stakeholders and business partners. It is therefore important to measure these value provisions of IS at this stage.

e. *Competencies Development and Management*. During the course of performing asset lifecycle management activities, engineering organisations generate enormous amount of explicit as well as tacit knowledge. The knowledge thus generated provides an organisation with competencies in managing its assets. IS not only have the ability to capture and process this knowledge, but can also facilitate knowledge sharing among organisational stakeholders. However, in order for this to happen it is important to find the fit between the social and technical systems in the organisation, since competencies development depends upon the functional/technical knowledge, as well as cultural, social, and personal values.
f. *Organisational Responsiveness*. Functional integration and a consolidated view of the asset lifecycle facilitate the asset managing organisation in responding to the internal as well as external changes. IS play an important role in materialising such responsiveness, due mainly to their ability to provide asset lifecycle profiling from financial and non financial perspectives. These value assessments aid the organisation in making decisions, such as asset redesign, retirement, renewal, as well as cost benefits of service provision and asset operation, and assessments of market demands. Nevertheless, the fundamental requirements in producing these value assessments are the availability integrated and quality information that allows for an integrated view of asset lifecycle though maintaining the asset lifecycle learnings.

This framework enables action oriented learning as it highlights the gaps between the existing and desired levels of performance, thereby necessitating the need for corrective action through (re)investment in right technology and skills, and acceptance of the change in the organisation. The evaluation thus provides triggers for continuous improvement regarding IS employed for asset design, operation, maintenance, risk management, quality management, and competencies development for asset lifecycle management. However, in order for that to happen a comprehensive approach is suggested as shown in figure 2 below. This approach suggests that the framework be applied to the four dimensions of IS in a systematic way, where perception of IS suitability and fitness of purpose that different stakeholders attach to it is assessed first. This feeds into an objective evaluation to assess the fit between the processes and technology in terms of the systems matching the information requirements. This provides input to contextual evaluation, where the emphasis is on measuring the four dimensions according to the prevailing operational, cultural, and social environment. After having done that, evaluations into maturity of technical architecture as well as businesses processes could be made. Evaluation in this way becomes a longitudinal study, however it ensures that it covers soft as well as hard aspects of IS, i.e., role of IS from simple data acquisition to enabling an organisation environment conducive to learning.

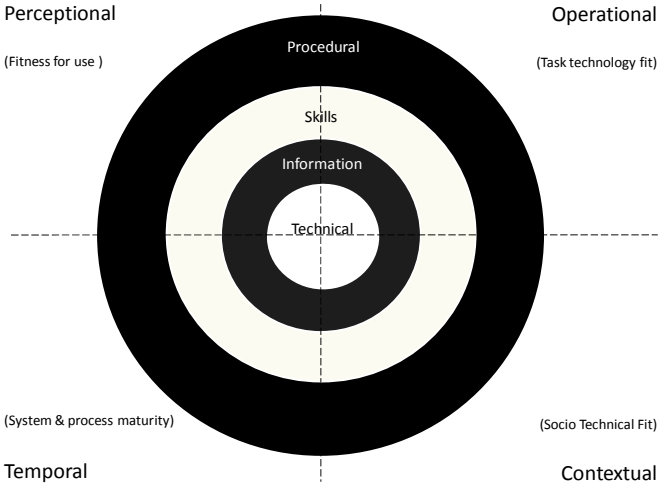


Fig. 2. Evaluation Perspectives

In order to provide an assessment of IS utilised in asset management using the proposed methodology, a stepwise approach is needed. In this approach, the framework proposed in figure 1 will be applied to each quadrant of the evaluation perspectives in figure 2. Starting from the perceptional quadrant in a clockwise direction the evaluation exercise will move towards temporal quadrant. In each quadrant the objective is to assess the technical, information, skills, and policy/procedural capability towards enabling value profiles of IS such as fitness of use, task technology fit, socio technical fit, and time system and process maturity in terms of their lifecycle. This framework could be applied qualitatively as well as quantitatively in each quadrant. Qualitative evaluation will require in-depth interviews and analysis of the process in each perspective of the framework illustrated in figure 1. Quantitative assessment includes quantitative assessment of processes under each perspective for dimensions of IS, i.e. information availability, information quality, people's skills, technology on a predetermined scale. This information could be collected though surveys and analysed with the help of Analytic Hierarchy Process and Multi-Attribute Utility theory.

6 IS for Asset Management Evaluation

Enacting appropriate methodologies, techniques, and tools for evaluation provide the rational underpinnings between the evaluation measures and the effectiveness of evaluation. Due consideration to this relationship is important, for the fact that IS implementation has a direct relationship with organisational context, human behaviour, and other structures developed around IS. Choice of evaluation method and tools needs to be comprehensive enough to encompass all these issues. IS enable business processes at each stage of an asset lifecycle, and also help in shaping the organisational infrastructure and social environment. IS, thus, have the ability to transform an asset managing organisation into a learning organisation through

facilitating organisational learning at each stage of the asset lifecycle. Evaluation of IS, therefore, needs to have a broad horizon and should account for assessment of their true value profile. Nevertheless, there are some conceptual and operational issues posed in enacting a robust evaluation methodology, which makes it essential that asset managing organisations take complete stock of the existing operational, social, and cultural environment of the business to establish comprehensive sets of evaluation dimensions and associated criteria. Furthermore, asset management requirements from IS illustrate that their implementation translates strategic objectives into action; align organisational infrastructure and resources with IS; provide integration of lifecycle processes; and informs asset and business strategy through value added decision support. However, the fundamental element in achieving these objectives is the quality of alignment of technological capabilities of IS with the organisational infrastructure. The generative learning based framework to evaluate the performance of IS utilised for asset proposed in this research allows asset managers to take stock of the existing IS capabilities and process maturity, as well the role that IS play in maturity of the organisational social environment. Using the proposed methodology will allow for the identification of right technological investment that satisfies the need pull as well as the change management strategies in allied areas that impact and are impacted by the introduction of technology. The generative learning based learnings provide for the continuous improvement of the asset management regime as well as the enabling IS infrastructure.

References

1. Davis, S., Albright, T.: An investigation of the effect of balanced scorecard implementation in financial performance. Management Accounting Research 15(2), 135–153 (2004)
2. Ittner, D., Larcker, D.F., Randall, T.: Performance implications of strategic performance measurement in financial services firms. Accounting Organisation and Society 28(7/8), 715–741 (2003)
3. Neely, A., Kennerley, M., Martinez, V.: Does the balanced scorecard work: an empirical investigation. In: Proceedings of Performance Measurement Association Conference (July 2004)
4. Braam, G.J.M., Nijssen, E.J.: Performance effects of using the balanced scorecard: A note on the Dutch experience. Long Range Planning 37(4), 335–349 (2004)
5. IIMM, International Infrastructure Management Manual, Association of Local Government Engineering NZ Inc., National Asset Management Steering Group, New Zealand, Thames (2006) ISBN 0-473-10685-X
6. OALD, 'The Oxford Advanced Learner's Dictionary, 7th revised edn. Oxford University Press, Oxford (2005) ISBN: 0194316491
7. Amadi-Echendu, J.E.: The paradigm shift from maintenance to physical asset management. In: Proceedings of 2004 IEEE International Engineering Management Conference, vol. 3, pp. 1156–1160. IEEE, Austin TX (2004)
8. Mitchell, J.S., Carlson, J.: Equipment asset management – what are the real requirements? Reliability Magazine, 4–14 (October 2001)
9. Haider, A.: Information systems based engineering asset management evaluation: operational interpretations. Phd Thesis, University of South Australia, Adelaide, Australia (2007)

10. Haider, A., Koronios, A., Quirchmayr, G.: You Cannot Manage What You Cannot Measure: An Information Systems Based Asset Management Perspective. In: Mathew, J., Ma, L., Tan, A., Anderson, D. (eds.) Proceedings of Inaugural World Congress on Engineering Asset Management, Gold Coast, Australia, July 11-14 (2006)
11. Rondeau, E.P., Brown, R.K., Lapides, P.D.: Facility Management. John Wiley & Sons, Hoboken (2006)
12. Konradt, U., Zimolong, B., Majonica, B.: User-Centred Software Development: Methodology and Usability Issues. In: Karwowski, W., Marras, W.S. (eds.) The Occupational Ergonomics Handbook, USA. CRC Press, Boca Raton (1998)
13. Orlikowski, W.J., Barley, S.R.: Technology and institutions: what can research on information technology and research on organizations learn from each other? MIS Quarterly 25(2), 245–265 (2001)
14. Walsham, G.: Making a World of Difference IT in a Global Context. John Wiley, Chichester (2001)
15. Farbey, B., Land, F., Targett, D.: Moving IS evaluation forward: learning themes and research issues. Journal of Strategic Information Systems 8, 189–207 (1999)
16. Neely, A.D., Gregory, M.J., Platts, K.: Performance Measurement System Design: A Literature Review and Research Agenda. International Journal of Operations & Production Management 15(4), 80–116 (1995)
17. Tangen, S.: Performance measurement: from philosophy to practice. International Journal of Productivity and Performance Management 53(8), 726–737 (2004)
18. Bijker, W.E., Law, J.: Shaping Technology/Building Society: Studies in Sociotechnical Change. MIT Press, Cambridge (1992)
19. Hayes, N., Walsham, G.: Competing interpretations of computer-supported cooperative work in organizational contexts. Organization 7(1), 49–67 (2000)
20. Orlikowski, W.J.: The Duality of Technology: Rethinking the Concept of Technology in Organizations. Organization Science 3(3), 398–427 (1992)
21. Willcocks, L.P., Lester, S.: In search of information technology productivity: Assessment issues. Journal of the Operations Research Society 48, 1082–1094 (1997)
22. DeLone, W.H., McLean, E.R.: Information systems success: The quest for the dependent variable. ER 1992 3(1), 60–95 (1992)
23. Teubner, R.A.: The IT21 Checkup for IT Fitness: Experiences and Empirical Evidence from 4 Years of Evaluation Practice. In: Becker, J., Backhaus, K., Grob, H.L., Hoeren, T., Klein, S., Kuchen, H., Muller-Funk, U., Thonemann, U.W., Munster, Vossen, G., Munster (eds.) European Research Center for Information Systems No. 2 (2005) ISSN 1614-7448

Fast Unsupervised Classification for Handwritten Stroke Analysis

Won-Du Chang and Jungpil Shin

Graduate School of Computer Science and Engineering, University of Aizu
Aizu-Wakamatsu, Fukushima, Japan
{d8081104,jpshin}@u-aizu.ac.jp

Abstract. This paper considers the unsupervised classification of handwritten character strokes in regards to speed, since handwritten strokes prove challenging with their high and variable dimensions for classification problems. Our approach employs a robust feature detection method for brief classification. The dimension is reduced by selecting feature points among all the points within strokes, and thus the need to compare stroke signals between two different dimensions is eliminated. Although there are some remaining problems with misclassification, we safely classify strokes according to handwriting styles through a refinement procedure. This paper illustrates that the equalization problem, the severe difference in small parts between two strokes, can be ignored by summing all of the differences via our method.

Keywords: Handwritten character classification, Time series data, Self-organized classification, Dynamic time warping, Dynamic programming.

1 Introduction

Since pen-tablet devices have enabled natural handwriting as input devices for a computer, the research on handwritten character analysis has become increasingly popular in recent decades. With many different facets of handwriting analysis, such as handwritten character recognition, identification, verification, and synthesis, researchers have developed their own analysis techniques and employed them for their purposes.

Although there are many issues that must be considered for handwritten character analysis, classifying characters automatically is one of the most important issues because most analyses start with finding similar and different patterns among the data.

However, classification is not an easy subject because of the problem of high and variable dimensions—the handwritten characters consist of a number of sequential points (See Fig. 1). In the point of traditional pattern classification, we can find two consequential problems: 1. Heavy time costs, 2. *Different order systems*: two points of the same order from two different signals may have different meanings (See Fig. 2). The second problem is especially severe in handwriting classification because it occurs even when the dimensional size is the same.

A conventional solution to this problem is dynamic time warping (DTW), which searches for the best corresponding points between two signals through an exhaustive search [7, 8]. The problem of DTW is that it suffers from high time complexity.

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 918–927, 2009.
© Springer-Verlag Berlin Heidelberg 2009



Fig. 1. Samples of handwritten characters (number 2). The numbers of points (denoted as circles) are different each other.

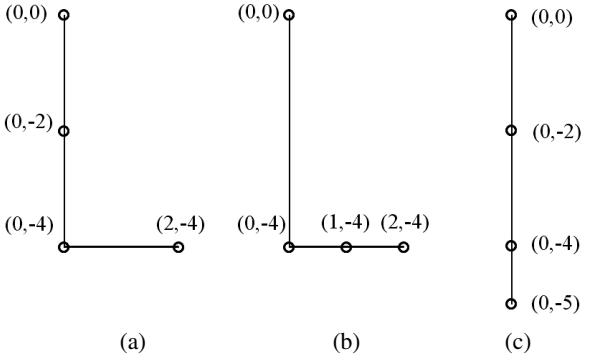


Fig. 2. Illustration of the problem of *different order systems*. The Euclidean distance between (a) and (b) is 3, whereas the distance between (a) and (c) is $\sqrt{5}$, although (a) and (b) have more similar shapes. It is because each order of signals has different meanings.

Although it generates good results for character classification, it is not practical for use with large databases because of the complexity.

This paper reports on a rapid classification method for handwritten characters, especially for Chinese characters in Japan (Kanji). Kanji consists of more than 30,000 strokes, and fast classification is severely required to analyze all of the strokes. Our specific goal for our classification is to classify handwritten strokes having similar writing styles.

In the following sections, we survey related literature in Section 2, and describe our new method in Section 3. Classification results are shown in Section 4, and conclusions appear in Section 5.

2 Related Work

The subject, unsupervised classification of handwritten characters, can be easily found in literature, usually composed of two parts: distance calculation and categorization. Most of the research has focused on the calculation of distances because conventional methods could be employed for the categorization if the calculated distances are real numbers.

The traditional approaches to the former are referred to as the hidden Markov model (HMM) and DTW, and they offer global optimal solutions if their systems are

designed well. HMM has been employed in conjunction with K-means [6], and dendrogram [1]; DTW has been employed with CH index [10], and dendrogram [9]. Besides, hybrid systems of HMM and DTW was proposed by [3, 5].

Although these researches have produced solutions to support unsupervised clustering of handwritten characters, their methods do not solve the problem of high time complexity, which is mentioned in the introduction. Since HMM and DTW search for adequate corresponding points between two signals, they suffer from $O(n^2)$ time complexity where n is the number of points of the signal.

Through our survey of related literature, one exception, which does not suffer high time complexity, is the re-sampling of handwriting signals proposed by Vuurpijl & Schomaker [11]. Although their distance measure is fast for calculating Euclidean distances, it completely ignores the problem of *different order systems*, and thus its limitation is obvious.

Another problem with conventional distance measures is the 'Equalization' problem. As illustrated in Fig. 3, a severe difference with a small part of a stroke can be easily lost by summing all the differences at each pair of corresponding points. Consequently, it is difficult to distinguish two strokes which have long common parts but short different parts.

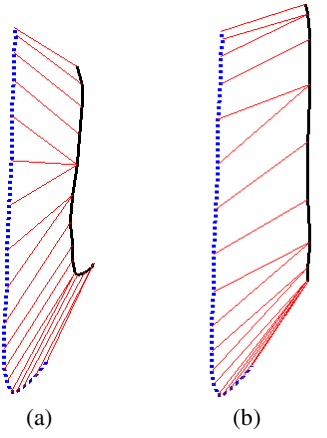


Fig. 3. DTW results among three handwritten strokes. A dotted line stroke is compared to two other strokes (in black) and their distances are calculated, after the corresponding points (connected with thin lines) are found by DTW. The Euclidean distances of (a) and (b) are 69.09 and 67.76, relatively. Consequently, these three strokes are grouped altogether or separated by each other according the threshold.

3 Method

Japanese Kanji characters, being used in common, consist of approximately 3,000 characters, or 30,000 strokes. Our goal is to analyze the strokes by finding similar strokes, eliminating the 'Equalization' and the *different order systems* problem with high speed. To overcome the problems, our method processes handwritten characters in feature point units. The method is composed of three steps: feature extraction and first level clustering, second level clustering using simplified ART2, and then refinement.

3.1 Level 1: Feature Extraction

We first extract feature points taking the 'perceptually importance point' recommended by Brault & Plamondon [2]. Since this method extracts one feature point for a single curve, similar strokes usually have the same number of points. At this level, all the strokes are divided in accordance to the number of their feature points (See Fig. 4 for samples of handwritten characters and selected feature points). Fine classification is done in the second level of clustering, and then errors are refined in the last step. Since only a single comparison is required to classify a pattern in this level, the time complexity of this level is *O(n)* for the number of patterns.

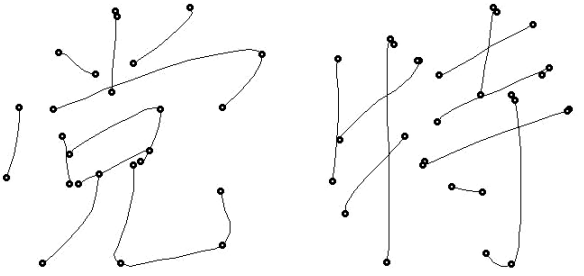


Fig. 4. Two handwritten characters with extracted feature points. The feature points are illustrated with white circle.

3.2 Level 2: Simplified ART2

After a coarse clustering of strokes according to the number of feature points, the resulting clusters are delicately classified at each cluster of the first level. Since the strokes have the same number of feature points within the groups (the same dimensional size), the general classification method can be easily employed. In this paper, a simplified ART2 algorithm [4] is selected and employed for this purpose because of its simplicity and speed. The time complexity of the algorithm is $O(m \cdot n)$ where m is the number of iteration and n is the number of patterns. When $m \ll n$, it could be rewritten as $O(n)$.

Unlikely K-means, it does not require the number of clusters in advance but requires tolerance thresholds. The algorithm, which is adapted for polar-coordinates, is as follows:

1. Set thresholds $\theta^{direction}$ and θ^{length} to define the sensitivity, define M for the total iteration number.
2. For each stroke $\{S^k \mid 1 \le k \le N\}$, where N is the total number of the strokes and $dist(C^i,S^k)$ is the distance between i^{th} centroid and S^k,
   2.1. Find C^t, where
   $$t = \arg\min_i \{dist(C^i,S^k)^{direction} + dist(C^i,S^k)^{length} \mid dist(C^i,S^k) < (\theta^{direction},\theta^{length})\}.$$
   2.2. If t=NULL, create a new cluster and set its centroid with S^k if no cluster is found.

2.3. If C^t is found, include S^k to C^t and update the centroid of C^t calculating the average of vectors included.

3. Repeat step 2 M times, and stop if there is no change of clusters.

In this algorithm, we used the polar-coordinates of (direction, length) instead of *xy*-coordinates, to differentiate the thresholds for the angle and the length of a segment.

In addition to the above algorithm, we propose a distance measure between a stroke and cluster centroid for polar coordinates, since the algorithm needed to calculate the closest cluster of a stroke and Euclidean distance cannot be applied to polar coordinates because of the scale difference. The distance between a stroke S and centroid C is defined as follows:

$$dist(C,S)^{direction} = \{\sum_{p=1}^{n} \left\| S_p^{direction} - C_p^{direction} \right\|\} / (n \cdot \pi), \tag{1}$$

$$dist(C,S)^{length} = \sum_{p=1}^{n} \left\| S_p^{length} / (2 \cdot l) - C_p^{length} \right\|, \tag{2}$$

where l is the total length of the stroke, and

$$dist(C,S) = \{dist(C,S)^{direction} + dist(C,S)^{length}\} / 2. \tag{3}$$

As it is shown in Equation (1) and (2), the distances are normalized linearly so that they have the same scales, as the maximum distances approach *1.0*.

To classify strokes according to a writer's handwriting style correctly, we designed Equation (2) concerning the ratios among segments of a stroke, where the segment means a part of a stroke between two feature points.

3.3 Refinement

The last step of our method is to correct the false-assigned strokes from clusters. The need for refinement occurs at the first step, which divides strokes according to the number of feature points.

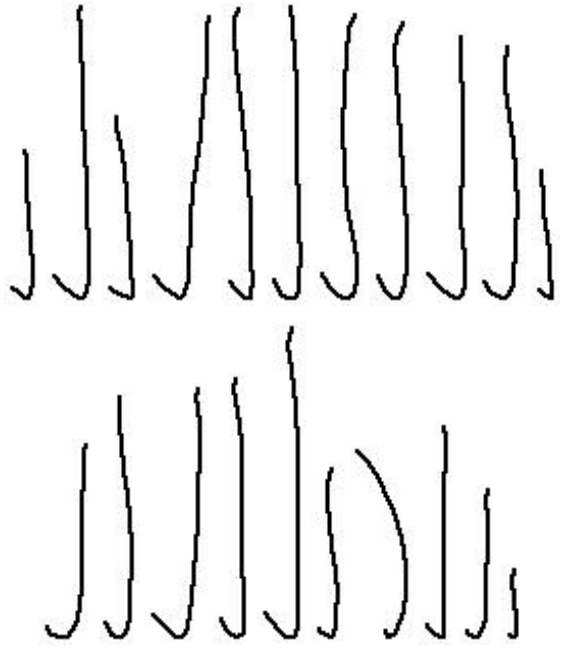
Although the number of 'perceptually important points' is very stable, a stroke should be checked for a cluster which has a different number of feature points, since a number of similar strokes generates a different number of feature points. Each point in the clusters are compared to the other clusters of having one more point after adding one more feature point to itself. We can compare a stroke over the clusters of the different numbers of feature points, and re-assign a stroke into a more adequate cluster. Since the refinement time complexity is only *O(n)* for the number of patterns, it can be employed for the entire classification procedure without burden.

4 Results

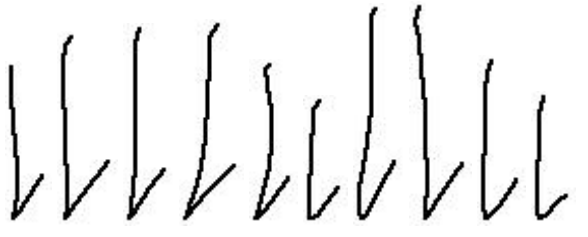
We applied the proposed method to a set of handwritten characters in numbers, English alphabets and Japanese Kanji. We classified 32,240 strokes from 3,118

characters, which were written by a writer. $\theta^{direction}$ and θ^{length} are set to π/12 (15°) and 0.25. Fig. 5 and 6 show the selected seven groups of clustering results among 2,632 groups.

Fig. 5 shows the classification results of simple horizontal lines and the horizontal lines having a hook. The existence of a hook should be considered important for distinguishing different handwriting styles, especially Japanese Kanji. One of the most important considerations for Kanji analysis is that Kanji consists of many straight lines and simple curves having partial differences in the strokes. Our system successfully classified the strokes according to the existence of hooks as it is shown.

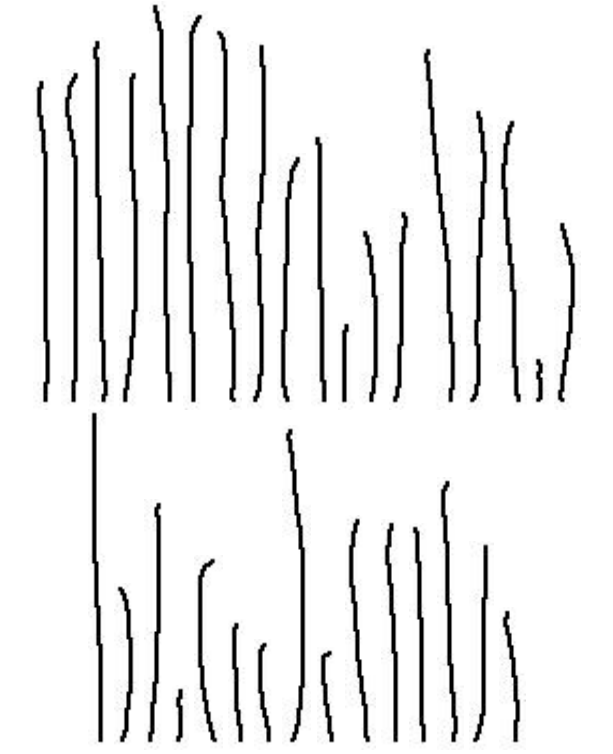


(a)



(b)

Fig. 5 (a-b). Two groups (c-d) of similar strokes after the proposed classification



(c)



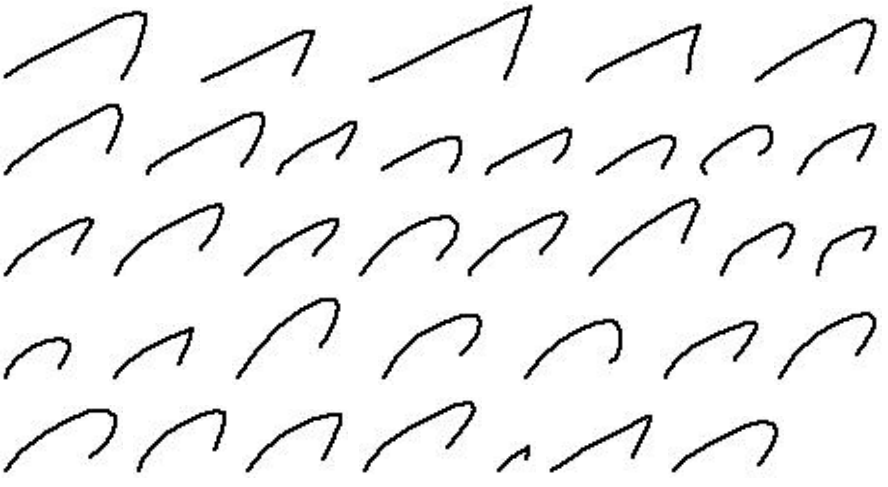
(d)

Fig. 5 (c-d). Two groups (c-d) of similar strokes after the proposed classification

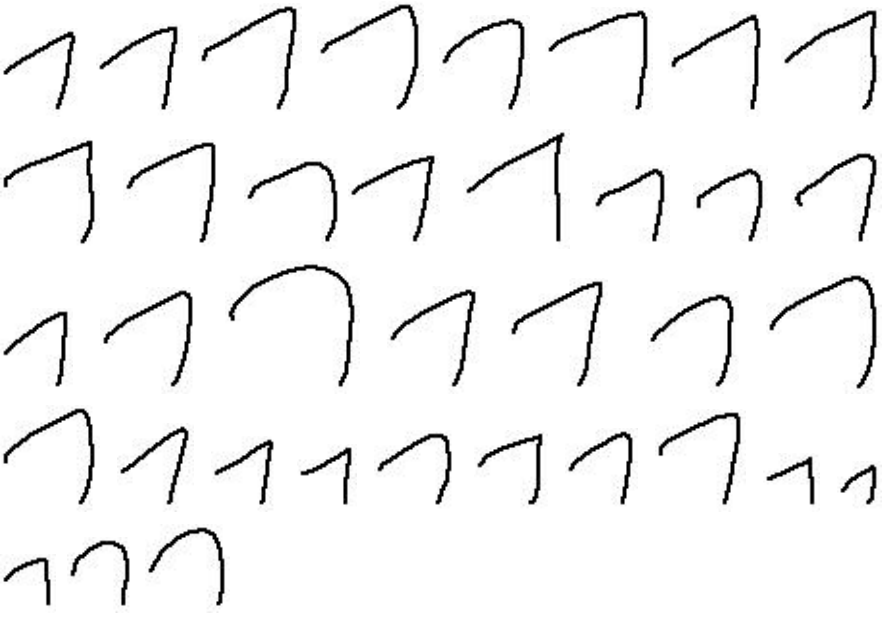
Fig. 6 describes the feature of our classification method. Strokes in various sizes are clustered together through the result, since our method tries to find similar shapes while ignoring the size. Strokes having similar/different ratios among their segments are successfully classified if the directional angles between segments are similar (See Fig. 6(b) and (c)).

Through these results, we observed two problems with this method. First, it does not distinguish any sharp corners from strokes with round corners (See Fig. 6(c). And since it uses feature points only to distinguish strokes, any detail of roundness is ignored through the classification (See the third strokes of Fig. 5(d)). Another problem

with our system is that it does not offer detailed hierarchical structures. However, we think these problems can be solved by employing an additional clustering procedure using entire points within each group. Time complexity would not increase much, as the strokes that need to be compared are limited within a group.

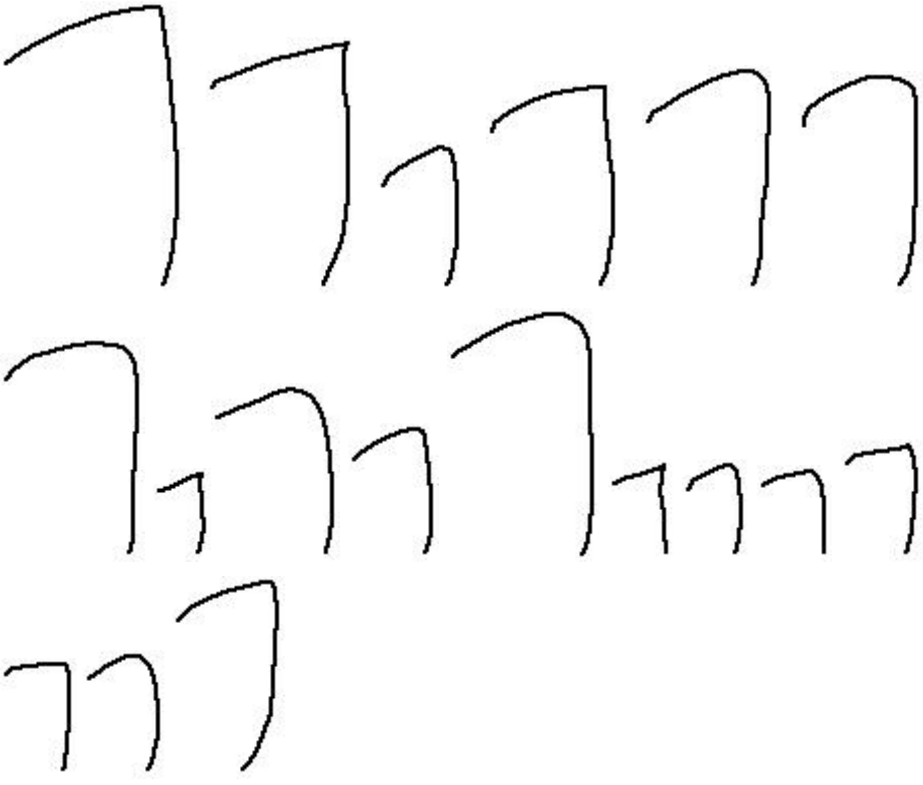


(a)



(b)

Fig. 6 (a,b). Two groups (a, b) of similar strokes after the proposed classification



(c)

Fig. 6 (c). A group (c) of similar strokes after the proposed classification

5 Conclusions

In this paper, we have described a fast classification method for handwritten strokes, using a robust feature point detector and refinement procedure. Although most of the current algorithms report $O(n^2)$ time complexities, our method achieved $O(n)$ time complexity by reducing dimensions and dividing initial groups according to the size of the dimensions.

Furthermore, our method could distinguish partial differences while avoiding equalization effects. Since the method removed unimportant points, we could successfully classify the strokes having partial differences through the experiments.

Although there are some limitations in classifying strokes according to the roundness of their curves, the proposed method solved the problem of classification according to the existence of hooks in the low time complexity of $O(n)$. We also think this limitation can be overcome soon in our future research.

Acknowledgements. We would like to thank Dr. Thomas Orr, Director of the Center for Language Research at the University of Aizu, for his helpful comments during the development of this paper.

References

1. Bahlmann, C., Burkhardt, H.: The Writer Independent Online Handwriting Recognition System Frog on Hand and Cluster Generative Statistical Dynamic Time Warping. IEEE Transactions on Pattern Analysis and Machine Intelligence 26(3), 299–310 (2004)
2. Brault, J.J., Plamondon, R.: Segmenting Handwritten Signatures at Their Perceptually Important Points. IEEE Transactions on Pattern Analysis and Machine Intelligence archive 9, 953–957 (1993)
3. Hu, J., Ray, B., Han, L.: An Interweaved HMM/DTW Approach to Robust Time Series Clustering. In: Proceedings of the 18th International Conference on Pattern Recognition, vol. 3, pp. 145–148 (2006)
4. Kim, B.-H., Koo, K.-M., Park, Y.-M., Cha, E.-Y.: 'A Study on Quantization Method Using ART2 for Contents-Based Image Retrieval' (in Korean), in proceedings 22nd Conf. of Korea Information Processing Society, vol. 11(2) (2004)
5. Oates, T., Firoiu, L., Cohen, P.R.: Clustering Time Series with Hidden Markov Models and Dynamic Time Warping. In: Proceedings of the IJCAI 1999 Workshop on Neural, Symbolic and Reinforcement Learning Methods for Sequence Learning, pp. 17–21 (1999)
6. Perrone, M.P., Connell, S.D.: K-means clustering for Hidden Markov Models. In: Proceedings of the Seventh International Workshop on Frontiers in Handwriting Recognition, pp. 229–238 (2000)
7. Sakoe, H.: A Generalized Two-Level DP-Matching Algorithm for Continuous Speech Recognition. IEICE Transactions E65-E(11), 649–656 (1982)
8. Shin, J.: On-line Cursive Hangul Recognition that Uses DP Matching to Detect Key Segmentation Points. Pattern Recognition 37(11), 2101–2112 (2004)
9. Vuori, V., Oja, E.: Analysis of Different Writing Styles with the Self-Organizing Map. In: Proceedings of the 7th International Conference on Neural Information Processing, vol. 2, pp. 1243–1247 (2000)
10. Vuori, V., Laaksonen, J.: 'A Comparison of Techniques for Automatic Clustering of Handwritten Characters'. In: The proceedings of the 16th International Conference on Pattern Recognition, pp. 168–171 (2002)
11. Vuurpijl, L., Schomaker, L.: Finding Structure in Diversity: a Hierarchical Clustering Method for The Categorization of Allographs in Handwriting. In: Proceedings of the 4th International Conference on Document Analysis and Recognition, pp. 387–393 (1997)

Interfaces for All: A Tailoring-Based Approach

Vânia Paula de Almeida Neris and M. Cecília C. Baranauskas

Institute of Computing, IC – UNICAMP, Campinas – São Paulo, Brazil
{neris,cecilia}@ic.unicamp.br

Abstract. Following the precepts of Universal Design, we must develop systems that allow access to software applications without discrimination and making sense for the largest possible audience. One way to develop Interfaces for All is to offer users the possibility of tailoring the interface according to their preferences, needs and situations of use. Tailorable solutions already present in some interactive systems do not consider the diversity of users, as they do not include illiterates and non-expert users, for example. The development of systems to be used for all requires a socio-technical vision for the problem. In this paper we present and discuss the first results of a work based on the reference of Organizational Semiotics and Participatory Design to elicit users' and system's requirements, and design a software solution with the direct participation of those involved, under the design for all principles.

Keywords: Universal Design, User Interfaces, Tailoring, Organizational Semiotics, Participatory Design.

1 Introduction

Nowadays, many services have been offered to the population through computers and the Internet: bills payment, communication with friends and institutions, searching for a job, among others. Besides the reduction in computer prices, the dissemination of cell phones and the implementation of telecenters and Internet cafes, many people still do not benefit from these services, especially in developing countries. One of the problems is that the way user interfaces are designed today, do not favor the interaction of the population in general by failing to consider the different users needs in the population.

Following the precepts of Universal Design or Design for All [24], we must develop systems that allow access to knowledge without discrimination and that make sense for the largest possible number of users according to their different sensory, physical, cognitive and emotional abilities.

Eliciting the interaction abilities present in population is essential to develop systems that can be used by the largest extension possible of users. However, the interaction needs may change over time and in different scenarios of use. Thus, in addition to offering various forms of interaction, systems should be adjustable so that they could accommodate the non-anticipated needs and the users' evolution [4].

Tailoring is the expression used in literature to define the activity of changing a computer application according to its context of use [10]. Tailoring involves the

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 928–939, 2009.
© Springer-Verlag Berlin Heidelberg 2009

concept of "design for change", offering the flexibility of being adapted to different organizational contexts or not anticipated situations of use, or those that have changed.

To offer systems that allow tailoring requires a logical structure to manage the possibilities of changing and an architecture that allows altering the system at the time of use. Current research on tailoring have focused on technical issues to enable adjustable applications (e.g. [13; 25]). However, improvements in aspects of implementation have not resolved the issues of design to make this technology accessible to all, including people with disabilities, elderly, illiterates and non-expert users.

We believe that the development of systems that intend to be for all requires a socio-technical vision for the problem. E-Cidadania is an ongoing Project in Brazil, in which we are experimenting with the design for all in a demanding context of the population of users involved [5]. To deal with this challenge we have chosen the reference of Organizational Semiotics [22; 12] (OS), allied to methods and techniques of Participatory Design [20] and Inclusive Design [3; 14] to clarify the problem, model the context and elicit users' and system's requirements with the direct participation of those involved. This work presents the approach we are using in the context of e-Cidadania Project to build interfaces for all tailorable to everyone.

The paper is organized as follows: Section 2 presents some related works and a summary of the theoretical reference; Section 3 describes the approach, considering 3 main phases and exemplifies each of them. Section 4 discusses some lessons learned and Section 5 presents some conclusions.

2 Background Work and Theoretical Reference

According to the Center for Universal Design from the State University of North Caroline-USA, the Design for All is the design of products and environments to be usable by all people, to the greatest extent possible, without the need for adaptation or specialized design. The design of Interfaces for All aims at addressing efficiently and effectively the problems arising from the users' different interaction abilities [23].

Connell et al [3] have defined some principles and guidelines for developing universal products. They are related to the equitable use, meaning that the design should be useful and marketable to people with diverse abilities; design should be flexible to accommodate a wide range of individual preferences and abilities, products should be ease to use and intuitive, the design should communicate necessary information effectively to the user, regardless of ambient conditions or the user's sensory abilities, among others.

The development of Interfaces for All is still a challenge as design problems persist even if we consider some particular users groups (cf. [18]). In some cases, only the use of assistive technologies (such as screen readers or automatic translators) and adherence to the recommendations of accessibility found in the literature are not sufficient for the effective interaction of these users [14]. Melo and Baranauskas [14] show the need to bring these people into the design process to understand their needs and to design with and for them.

One way of developing Interfaces for All is to offer to the users the possibility of tailoring the interface according to their preferences, needs and situations of use.

However, tailorable interfaces already present in some interactive systems have not shown effectiveness [25]. In general, the interfaces do not clearly communicate the opportunity to be tailored and, when they do, they require skills that non-sophisticated users do not have. Thus, it is necessary to investigate new approaches for the design of tailorable systems focusing on Interfaces for All.

Literature shows some works that have used the Participatory Design discipline to design tailorable systems [11, 4]. The contexts experienced in such works were related to business environments, with focus on well-established communities of practices and interaction requirements different from those considering illiterates and non-expert users. Moreover, the context of this research involves other kinds of differences, besides the issue of disability itself; it is necessary to know the different interaction requirements (social, cognitive, emotional etc.) that characterizes the target users. In this sense, we have chosen a theoretical reference that presents a socio-technical vision to the development of information systems, as shortly described following.

2.1 Organizational Semiotics

OS is a discipline that has roots in Semiotics applied to organizational processes. It studies the nature, characteristics, function and effect of information and communication within organizational contexts. An organization is a social system in which people behave in an organized manner conforming to a certain system of norms. These norms are regularities of perception, behavior, belief and value that are expressed as customs, habits, patterns of behavior and other cultural artifacts [22; 12].

By using Semiotics, the human-computer interaction can be understood through complex processes. Such processes, analyzed only according to the perspective of engineering, have been interpreted as purely syntactic phenomena. The analysis using Semiotics rescues the primary function of computer systems as vehicles of signs and supplies an adequate vocabulary to understand the relation between computer systems and other sign systems [16].

Stamper has proposed a set of methods to support the use of OS concepts for modeling information systems, named MEASUR - Methods for Eliciting, Analyzing and Specifying Users' Requirements [22]. Our approach to build Interfaces for All, builds on 3 MEASUR methods: Problem Articulation Method (PAM) – to identify the main topics related to the context, allowing a clear understanding of the problem; Semantic Analysis Method (SAM) – to focus on the agents and their pattern of behavior (named affordances) to describe the organization and its information system functions in ontology charts; Norm Analysis Method (NAM) - usually carried out on the basis of the result of SAM to specify the condition and constraints on the behaviors. The next section presents a practical application of these methods in the context of design for all at e-Cidadania

3 Building a Tailorable Application

The development of a technical system that intends to be inclusive and suitable for as many people as possible faces the challenge of eliciting different interaction requirements and designing proper user interfaces. Moreover, the construction of a tailorable

solution also faces the need of software architecture capable of managing the different interaction options. Figure 1 presents the main phases, and the related inputs, of our tailorable approach to the development of Interfaces for All. Next sub-sections present details of each main phase, which are exemplified by the activities conducted in the context of e-Cidadania.

E-Cidadania involves a multidisciplinary team investigating the relationship people establish in their informal communities organized around some special interests, how they use societal artifacts, including computational technology, aiming at the design and development of a social network system [5]. The team involves designers, software engineers, anthropologists, educators, people from the media area, developers and community leaders. From the community of prospective users, 15 representatives were invited, including weavers, hairdressers, maidservants, retirees, teachers from a pre-college school, telecenter monitors, government representatives among others.

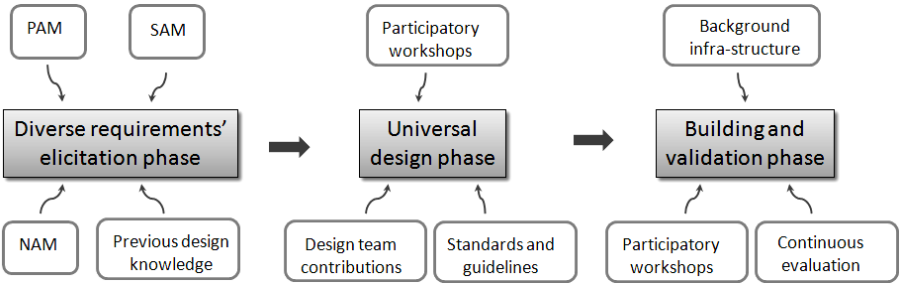


Fig. 1. A tailorable approach for building Interfaces for All

3.1 Gathering Requirements from the Diversity

Requirements elicitation is a fundamental phase in any development cycle. Moreover, while developing a universal design, requirements elicitation turns to be even more important. Besides the challenge of eliciting different interaction needs, the designer may be dealing with users s/he does not know much about. Interfaces that intend to be for all extrapolate the well-known frontiers of the office applications. Our approach to deal with these not well-known interaction needs is to bring these users to the design process.

Also, interfaces for all ask for an elicitation approach that considers more than just technical issues. MEASUR allows us to clarify the problem, elicit semantic information and define the responsibilities and related agents. With this clarified view of the context, we can determine which actions will be executed by the system. The definition of responsibilities is essential for tailorable systems once there are many agents with their different interaction needs that ask the system a different interaction behavior. Although the MEASUR methods can be applied in different orders, in our approach we have used PAM, SAM and NAM in this order.

In the e-Cidadania project, PAM was used in a workshop format with the Stakeholder Analysis Chart (see Figure 2a) and the Evaluation Framing Chart. The activities lasted three hours and they took place at the CRJ – Centro de Referência da Juventude (Youth Reference Center). Chairs were arranged in a semi-circle in front of

the artifacts hung on one of the walls. Post-its were distributed to the participants who would write their ideas on them and hand them to be posted on the artifacts (cf. [7]).

The Stakeholder analysis guide us to think about stakeholders that are directly responsible for the system – called actors, also about clients and suppliers, partners and competitors, as well as community and government interested or affected by the system. In e-Cidadania, 59 different stakeholders were mentioned, including housewives, elderly, people with disabilities, health agents, community leaders, neighborhood associations and religious institutions.

The Evaluation Framing Chart allows the elicitation and discussion of problems and issues the mentioned stakeholders would face, as well as ideas and solutions for these problems. With this chart, we intend to extract the main issues that should be considered while developing the system. For example, in e-Cidadania project, participants reported concerns related to low educational level and literacy proficiency of the prospective users. For these problems, they pointed out the use of audio-visual content and accessible vocabulary as possible solutions. Considering the universal design principles, these requirements can be supplied on a tailorable solution. Also, questions related to the environment and financial support were mentioned.

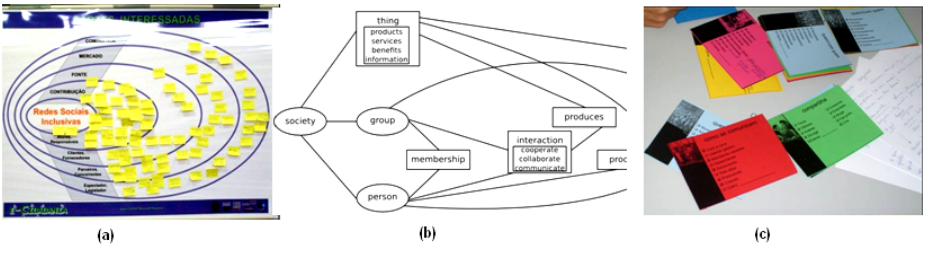


Fig. 2. (a) Stakeholder Analysis Chart. (b) Part of the Ontology Diagram. (c) Cards used in the second workshop.

The second method used in the elicitation phase was SAM. We applied SAM as it was originally proposed, with its four major phases (cf. [12]). However, for the first stage in the semantic analysis, which is problem definition, we used the description that participants of the workshop wrote about their concepts of an inclusive social network. From their definitions, the design team generated the affordances candidates, grouped them and drew an Ontology Chart for inclusive social networks. Figure 2b shows part of this Ontology Chart (for the complete chart, cf. [17]).

From Figure 2b it is possible to see that the root element "society" affords "person", "group" and "thing". "Person" and "group" afford "membership". This relation is important to represent the digital inclusion process. In this scenario, "group" represents any set of people, including that group that has access to information and communication through computers. This represents that any technical system that intends to support inclusive social networks should make "membership" possible which implies in important design issues regarding accessibility and universal design.

"Person" and "group" also afford "interaction". Furthermore "interaction" and "thing" afford "produces". These relations represent that interacting, in such modes as communicating, cooperating or collaborating, the groups are able to produce things that can be products, services or even information. These different interaction modes demand different functionalities to support them. For example, to allow communication, users should be able to share a system of signs, which will be possible only if the user interfaces make sense to each of them.

The next step in our approach was to elicit the different ways people interact, communicate, and collaborate, to cite some of the interaction modes. Our intention was to clarify the notions of responsibilities to define the tailorable behavior. A second workshop was proposed and conducted to elicit their norms and start application of the NAM. For this workshop, the original group of participants was extended with people chosen by their roles and activities performed in the target community, as a result of the stakeholders analysis of PAM.

For the second workshop, we have adapted a technique from the Participatory Design field known as "CARD" [15]. As in the use of the CARD technique, the participants were given cards and through them they were able to organize their ideas and present their storytelling experiences. Differently from CARD, new categories were created in order to capture information regarding the way they organize themselves in social networks from the resulting stories. The new categories were inspired on the elements of the Ontology Chart (built as a result of the SAM) and they arose from the statement of the question: "Who shares what with whom, when, how, where, using what and why?". Figure 2c shows some of the cards used. After the workshop, the design team worked on the analysis of each story and a group of norms was elicited. From the 21 stories reported, 37 norms regarding the community social dynamics were defined. Table 1 illustrates some norms obtained.

In the process of writing norms, the first step was to identify the main actions in each story, listed in the <action> column. Then, the responsible for the action was identified and put in the <agent> column. Subsequently, we analyzed the deontic operator, i.e. the information whether the agent must (as obligation), must not (as prohibition) or may (as permission) perform the action. Finally, we recognized the triggers, which were written in the <condition> and <state> columns [6].

It is important to notice that the set of norms represents what the community has been doing up to now. For example, from the last line of Table 1, we can see that announcements about the CIDARTE's parties are made by posters or face to face and also via email. The analysis task leads us to think about ways to support the announcements with tools to make the information accessible to more people (including the digitally illiterate). Also, from the third line of Table 1, it is possible to notice that different means of communication should be provided. As the students from Manga class communicate by drawing, the technical system should provide a tool for drawing or at least the possibility of uploading and publishing images. Moreover, from this set of norms, we can see that some users would like to draw, while others prefer to write or talk, indicating different functionalities the system should provide.

Table 1. Examples of norms from e-Cidadania project

whenever <condition>	if <state>	then <agent>	is <deontic operator>	to <action>
Always	before using some-one else´s knowledge	a person	must	ask permission to that person
During events or daily at CRJ	there are young people interested	teachers	may	offer Manga class using paper, pen and posters
Always	there is a former student of Herbert Souza course and s/he wants to cooperate with community	this former student	may	share knowledge about the course with current students at the school in person, using phone, paper and pen, television, board, computer/ internet.
Always	there is an event	CIDARTE coordinator	must	share with the group information about the event. S/He may use posters, face-to-face communication and also email.

The last input in the elicitation phase is the design team previous knowledge regarding the application domain. This knowledge may also include the users and the business rules, based on experience, literature review or even from design activities or internal workshops (for tailoring elicitation patterns cf. [1]). As outcome of the diverse requirements' elicitation phase, the design team can formalize the requirements in a format suitable to the specific types of requirements.

3.2 Designing a Universal Solution

In the proposed approach, after getting a requirements list, it is time to investigate how these functionalities can be offered following the precepts of the Design for All.

Universal solutions should provide the same means of use for all users: identical whenever possible; equivalent when not [3]. One way to achieve this is to define a conceptual model that should be followed while designing any part of the system. The formalization of the conceptual model should consider information from PAM (especially from the Evaluation Framing, where the main problems where pointed out), SAM (by the affordances from the Ontology Diagram) and NAM (by the norms that represent the expected behavior). Also, previous design knowledge should be considered. From the conceptual model it is possible to think about the different representations (interface elements or media) we may have on the interfaces.

A third workshop was conducted in the e-Cidadania project to explore user interface design solutions with the parties. We applied a Participatory technique called BrainDrawing [15], a method that allows a rough design of user interfaces through a cyclical brainstorming. In the BrainDrawing, each participant starts a drawing in one sheet of paper. After a short period of time, the participant gives his/her sheet to the next participant which will continue the actual drawing. Each drawing, at the end, is a fusion of ideas from everyone involved and each design is unique because it had a different beginning.

The workshop started with a brief statement describing one of the scenarios of use for the prospective system. The participants were organized into 5 groups. After the BrainDrawing, each group discussed the drawing results and get to one consensual solution that they presented to the other groups. During discussion, we could identify the essential interface elements and interaction styles that the inclusive social network system should offer.

Figure 3 shows 3 of the 5 consolidated designs. In the pictures, it is possible to see that the groups have chosen different navigational structures – Figure 3 (a) shows a linear menu while in (c) it is possible to see a circular menu. Also, there are different positions for some interaction areas – Figure 3 (c) shows the announcement in a central position and (b) shows the announcement area positioned on the left side. Differences appeared also in the way people would communicate. In one of the proposals, users could communicate writing messages (like in a chat) while in another, only a telephone number should be presented.

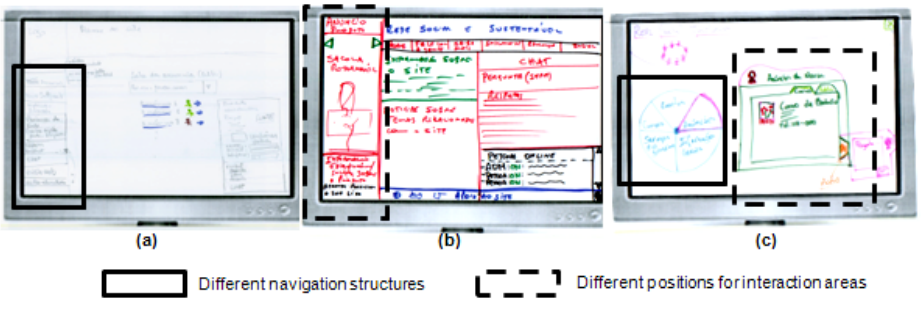


Fig. 3. Some design proposals obtained with the BrainDrawing technique

From the workshop it was possible to obtain many design ideas and also a refinement of the requirements. However, to obtain the universal design proposal, the design team has to work on the available design ideas. In this sense, another input in our approach is the Design team contributions.

Another important source of knowledge that contributes to the design phase is the group of Standards and guidelines related to accessibility (cf. http://www.w3.org/WAI; http://warau.nied.unicamp.br). A universal solution has to be accessible as a pre-requirement. Therefore, it is important to follow the recommendations and consider efficient assistive technologies and techniques (cf. [9]).

As outcomes of the design phase, the conceptual model can be formalized in a design rationale format, for instance. Interface design proposals can be represented by sketches or low fidelity prototypes.

3.3 Building and Evaluating the Solution

After obtaining the conceptual model and a proposal for the design of the user interfaces, it is possible to prototype the application. Considering software engineering principles, it is important to formalize all the information acquired aiming at the coding phase. At this point, Use Cases and System Sequence Diagrams can be specified (cf. [21]). However, offering universal interface solutions, providing different and

suitable forms of interaction requires an infra-structure which allows managing the changes and altering the system at the time of use. Literature shows some possibilities of infra-structures that can be applied (cf. [13; 25; 2]).

In e-Cidadania project, we are using the Bonacin's infra-structure because it also considers OS as a reference and proposes the use of norms to manage the possibilities of tailoring [2]. Figure 4a shows the architecture defined for tailoring-based solutions in the e-Cidadania project. The designer enters norms in a software application named norms editor. The NBIC (Norm Based Interface Configurator) receives the norm specification in Deontic logic, manages the norms persistence, and also transforms them into a platform specific language that can be interpreted by an inference machine on ICE (Interface Configuration Environment). Then, the ICE receives context information from the Tailoring Development Framework, evaluates the norms related to context by using an inference machine and returns to the framework an action plan with the changes to be done [2]. The framework works with a content management system, in e-Cidadania case - the Drupal, and makes available tailorable user interfaces. Figure 4b shows examples of interfaces with different interaction elements. One solution presents a linear menu while the other one provides a circular menu. Also, in the first one, information is accessible by text, while in the other one there is a space for a virtual actor that can speak or make signs.

In addition to the building of the design proposal, evaluation is also an important aspect to consider. In the context of e-Cidadania, evaluation is being considered in two moments: during participatory workshops, where some evaluation frameworks can be applied, as the Self Assessment Manikin (cf. [8]), and in a continuous on-line evaluation in which more longitudinal studies can be done. In continuous evaluation, expected results are the identification of user behaviors, learning curves, communication styles, etc. Relevant data to be captured are individual as well as group interactions; data can be captured using embedded tools that gather user statistics respecting the users' privacy [19].



Fig. 4. (a) Architecture proposed for tailoring in the e-Cidadania project. (b) Instances of tailorable interfaces.

4 Discussion and Lessons Learned

The development of Interfaces for All demands a clarified view of the problem and of the different interaction requirements present in the users population. From the stakeholders and problems/ solutions mentioned here, it is possible to see how PAM supports the elicitation of different stakeholders and between them, the diversity of users.

Further, Connell and others [3] indicate that during the development of a universal solution, designers should also incorporate considerations related to economics, engineering, culture, gender and environmental issues. The Evaluation Framing Chart supports the elicitation and discussion about these topics in a participatory way.

Moreover, the involvement of the different users is a crucial aspect in the proposed approach. In this sense, it is important to point out the need of providing a warm and non-intimidating environment for the workshops. Also, it is necessary to use an accessible vocabulary and open to everyone the opportunity to speak. For instance, in some of the definitions users wrote about inclusive social networks (that were used in SAM), their grammar mistakes did not prevent them to express a high level of maturity and consciousness regarding the topic.

From the elicited requirements, we could notice the need of using different media to make information accessible in a universal way. In addition, redundancy showed to be necessary for the universal design. For instance, for the interaction of the illiterate or with low literacy people it is possible to find in literature works that consider interfaces without text as a possible solution (cf. [18]). However, despite these interfaces allow users to access content by images and sounds, they do not provide the contact with the text, a key element in promoting the ability to read. User interfaces should be also considered as means of promoting the intellectual growth of the users.

Besides that, it is important to emphasize that the universal design solutions, when possible, should prepare the users to interact with other systems. This is a key aspect considering digital inclusion. Finally, Interfaces for All are related to the right to choose the interaction way which is more suitable for each user. In this sense, universal design solutions should always provide means to users benefit from technology despite any previous background.

5 Conclusions

This paper brought to discussion the problem of designing for a diversity of users competencies typical of contexts of digital divide. The complexity of the social scenario which includes people not familiar with technology suggests the need of approaches for requirements elicitation that traditional methods from Information Systems and Software Engineering fields do not reach. The paper described the approach we are investigating in the context of the e-Cidadania project, which brings prospective users to the design process and uses a theoretical reference that allows a socio-technical vision to the problem. The requirements elicitation, design and building phases were presented, exemplified and discussed.

The approach we proposed here is to build Interfaces for All, tailored to each one. By applying this approach in the e-Cidadania project we were able to identify issues that could be missed out in a strict technically-based approach (e.g. the needs of asking permission before using someone else's knowledge), especially regarding how to make the solution tailorable. Further work includes the evaluation of the tailorable behavior of the system to different types of social norms generated by the users.

Acknowledgements. This work is funded by FAPESP (#2006/54747-6) and by Microsoft Research - FAPESP Institute for IT Research (#2007/54564-1). The authors

also thank colleagues from NIED, InterHAD, Casa Brasil, CenPRA, IC-UNICAMP and IRC-University of Reading for insightful discussion.

References

1. Baranauskas, M.C.C., Neris, V.P.A.: Using Patterns to Support the Design of Flexible User Interaction. In: Jacko, J.A. (ed.) HCI 2007. LNCS, vol. 4550, pp. 1033–1042. Springer, Heidelberg (2007)
2. Bonacin, R., Baranauskas, M.C.C., Santos, T.M.: A Semiotic Approach for Flexible e-Government Service Oriented Systems. In: 9th ICEIS 2007. v. ISAS, pp. 381–386 (2007)
3. Connell, B.R., Jones, M., Mace, R., et al.: The Principles of Universal Design 2.0. Raleigh. The Center for Universal Design, NC State University (1997), http://www.design.ncsu.edu/cud/about_ud/udprinciples.htm
4. Costabile, M.F., Fogli, D., Fresta, G., Mussio, P., Piccinno, A.: Building Environments for End-User Development and Tailoring. In: Human Centric Computing Languages and Environments, pp. 31–38. IEEE Press, New York (2003)
5. e-Cidadania Project. Systems and Methods in the constitution of a mediated by Culture and Information Communication Technologies. FAPESP-Microsoft Research Institute (2006), http://www.nied.unicamp.br/ecidadania
6. Hayashi, E.C.S., Neris, V.P.A., Almeida, L.D.A., Miranda, L.C., Martins, M.C., Baranauskas, M.C.C.: Clarifying the dynamics of social networks: narratives from the social context of e-Cidadania. IC-08-030 (2008), http://www.ic.unicamp.br/publicacoes
7. Hayashi, E. C. S., Neris, V. P. A., Almeida, L. D. A., Rodriguez, L. C., Martins, M. C., and Baranauskas, M. C. C.: Inclusive social networks: Clarifying concepts and prospecting solutions for e-Cidadania. IC-08-029 (2008), http://www.ic.unicamp.br/publicacoes
8. Hayashi, E.C.S., Neris, V.P.A., Baranauskas, M.C.C., Martins, M.C., Piccolo, L.S.G., Costa, R.: Avaliando a Qualidade Afetiva de Sistemas Computacionais Interativos no Cenario Brasileiro. In: Proc. Workshop UAI. Porto Alegre. Brasil (2008c)
9. Hornung, H., Baranauskas, M.C.C., Tambascia, C.A.: Assistive Technologies and Techniques for Web Based eGov in Developing Countries. In: Proc. 10th ICEIS 2008. v. ISAS, pp. 248–255 (2008)
10. Kahler, H., Morch, A., Stiemerling, O., Wulf, V.: Computer Supported Cooperative Work. Journal of Collaborative Computing - CSCW 9, 1–4 (2000)
11. Kjær, A., Madsen, K.H.: Participatory Analysis of Flexibility. Communications of ACM 38(5), 53–60 (1995)
12. Liu, K.: Semiotics in information systems engineering. Cambridge University Press, Cambridge (2000)
13. Macías, J.A., Paternò, F.: Customization of Web applications through an intelligent environment exploiting logical interface descriptions. Interacting with Computers 20(1), 29–47 (2008)
14. Melo, A.M., Baranauskas, M.C.C.: An Inclusive Approach to Cooperative Evaluation of Web User Interfaces. Proc. 8th ICEIS 1, 65–70 (2006)
15. Muller, M.J., Haslwanter, J.H., Dayton, T.: Participatory Practices in the Software
16. Helander, M., Landauer, T.K., Prabhu, P. (eds.): Lifecycle. Handbook of HCI, 2nd edn., pp. 255–297. Elsevier Science, Amsterdam (1997)
17. Nadin, M.: Interface Design: A semiotic paradigm. Semiotica 69(3/4), 269–302 (1988)

18. Neris, V.P.A., Almeida, L.D.A., Miranda, L.C., Hayashi, E.C.S., Baranauskas, M.C.C.: Towards a Socially-constructed Meaning for Inclusive Social Network Systems. In: 11th ICISO (2009) (to be published)
19. Neris, V.P.A., Martins, M.C., Prado, M.E.B.B., Hayashi, E.C.S., Baranauskas, M.C.C.: Design de Interfaces para Todos – Demandas da Diversidade Cultural e Social. In: Proc. 35o. SEMISH/CSBC, pp. 76–90 (2008)
20. de Santana, V.F., Baranauskas, M.C.C.: A Prospect of Websites Evaluation Tools Based on Event Logs. In: Proc. HCIS 2008, IFIP WCC 2008, USA, pp. 99–104 (2008)
21. Schüler, D., Namioka, A.: Participatory design: Principles and Practices. L. Erlbaum Associates, USA (1993)
22. Sommerville, I.: Software Engineering, 6th edn. Addison-Wesley Pub. Co., Reading (2000)
23. Stamper, R.K., Althaus, K., Backhouse, J.: MEASUR: Method for Eliciting, Analyzing and Specifying User Requirements. In: Olle, T.W., Verrijn-Stuart, A.A., Bhabuts, L. (eds.) omputerized assistance during the information systems life cycle. ESP (1988)
24. Stephanidis, C.: User Interfaces for All: New perspectives into HCI. In: Stephanidis, C. (ed.) User Interfaces for All, Lawrence Erlbaum Ass., NJ (2001)
25. Trace: Universal Design Principles and Guidelines (2006), `http://trace.wisc.edu/world/gen_ud.html`
26. Wulf, V., Pipek, V., Won, M.: Component-based tailorability: Enabling highly flexible software applications. Journal of Human-Computer Studies 66(1), 1–22 (2008)

Integrating Google Earth within OLAP Tools for Multidimensional Exploration and Analysis of Spatial Data

Sergio Di Martino[1], Sandro Bimonte[2], Michela Bertolotto[3], and Filomena Ferrucci[4]

[1] University of Naples "Federico II", Napoli, Italy
[2] Cemagref, UR TSCF, 24 Avenue des Landais, 63172 Clermont-Ferrand, France
[3] University College Dublin, Belfield, Dublin 4, Ireland
[4] University of Salerno, Fisciano (SA), Italy
dimartino@na.infn.it, sandro.bimonte@cemagref.fr,
michela.bertolotto@ucd.ie, fferrucci@ unisa.it

Abstract. Spatial OnLine Analytical Processing solutions are a type of *Business Information Tool* meant to support a Decision Maker in extracting hidden knowledge from data warehouses containing spatial data. To date, very few SOLAP tools are available, each presenting some drawbacks reducing their flexibility. To overcome these limitations, we have developed a web-based SOLAP tool, obtained by suitably integrating into an ad-hoc architecture the Geobrowser Google Earth with a freely available OLAP engine, namely Mondrian. As a consequence, a Decision Maker can perform exploration and analysis of spatial data both through the Geobrowser and a Pivot Table in a seamlessly fashion. In this paper, we illustrate the main features of the system we have developed, together with the underlying architecture, using a simulated case study.

Keywords: Spatial OLAP, Data Visualization, Spatial Decision Support Systems, Spatial Data Warehouses.

1 Introduction

Current technologies for data integration are enabling enterprises to collect huge amounts of heterogeneous data in data warehouses. From a business point of view, these repositories can contain very precious, but often hidden, information that could benefit the competitiveness of an enterprise. *Business Information Tools*, and in particular *OLAP* (OnLine Analytical Processing) solutions aim at supporting the Decision Makers in discovering this concealed information, by allowing them to interactively explore these multidimensional repositories of information through some visual, interactive user interface. Indeed, the main strength of these solutions is the possibility to discover unknown phenomena, patterns and data relationships without requiring the user to master either the underlying multidimensional structure of the database, or complex multidimensional query languages. As a consequence, a crucial role for the success of OLAP solutions is played by the adopted visualization techniques, that should effectively support the mental model of the Decision Maker, in order to take advantage of the unbeatable human abilities to perceive visual patterns and to interpret them [1, 2, 13].

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 940–951, 2009.
© Springer-Verlag Berlin Heidelberg 2009

This is especially true when dealing with spatial information, where the analytical process can help a Decision Maker in identifying unexpected relationships and patterns between phenomena and the geographical locations where they took place. It is worth noting that to date increasingly more spatial data is being collected into data warehouses, thanks to the availability of powerful georeferring tools, such as GPS and GIS. [14] showed that about 80% of the data stored in databases integrates some kind of spatial information. It is clear that during the analytical process the spatial dimension should not just be treated as any other descriptive dimension but the spatial nature of data should be taken into account when developing specific visualization techniques.

Spatial OLAP (SOLAP) techniques aim to address this issue. SOLAP has been defined by Bedard as a "as a visual platform built especially to support rapid and easy spatiotemporal analysis and exploration of data following a multidimensional approach comprised of aggregation levels available in cartographic displays as well as in tabular and diagram displays" [3]. Thus, a spatial analysis process should be based on SOLAP operators that should be trigged using both traditional tabular representations of data, and geographical maps [4]. Indeed, interactive maps enhance analysis capabilities of pivot tables, since they permit to explore the spatial relationships of multidimensional data, by means of suitable *Geovisualization* techniques, i.e. advanced geospatial visual and interaction techniques supporting geographic datasets analysis to discover knowledge [8, 20].

In spite of the importance of this field, to the best of our knowledge, to date very few tools have been developed that integrate OLAP and geovisualization techniques (see Section 2). In any case, they suffer from different drawbacks, which can be summarized as follows:

1. They use 2D maps. This could be enough in some contexts, but it is recognized in literature that for spatial analysis, 3D can greatly enrich analysis capabilities. Indeed, 3D displays help user in orientation and provide a more natural description of landforms and spatial aspects than traditional 2D displays (see, for example, [11]), which is fundamental for detecting and understanding geo-spatial phenomena [19].
2. They are not ready to easily integrate external data sources, as they usually rely on proprietary technologies. This is a main drawback, because an effective spatial analysis requires comparing the investigated phenomenon with the surrounding elements of interest on the land (e.g. roads, industries, cities, etc…). Thus the ability to import spatial information from other (potentially remote) data sources is fundamental for this kind of tools.
3. They do not permit high levels of personalization of the visual encodings of the (spatial) data. To match the Decision Maker mental model, it is important to provide the possibility to represent geographic data in different ways, since they could provide alternative insights on the data, and so reveal additional knowledge.
4. They are intended as traditional desktop applications: switching to web-based technologies could highly improve the spreading and the flexibility of these kinds of solutions.

In this paper, we propose a system we have developed, named Goolap that aims to address the above issues by suitably combining the facilities provided by a commonly used geobrower and a traditional OLAP system. In the following we present the technological solutions that allowed us to integrate in a single, web-based application, the

geobrowser *Google Earth* with a freely available OLAP server, *Mondrian*. The main advantage of this solution is to provide a web-based SOLAP environment, able to render in 3D spatial data stored in different data repositories, with a high degree of personalization of the visual encodings of the information.

The paper is structured as follows. Section 2 contains a brief recall of current related work on SOLAP. In Section 3 we describe the main features of the proposed system, introducing the user interface of the tool and an example of multidimensional analysis using a simulated case study. In Section 4 we describe the architecture and the technological solutions we adopted for the system we propose to support SOLAP tasks. Some final remarks and future work conclude the paper.

2 Related Work on SOLAP

Data warehouses are organized according to the *multidimensional model* [15]. In multidimensional models, facts are analyzed thanks to measures or indicators. The dimensions represent the axes of analysis; their members or instances are organized into hierarchies. This approach enables a Decision Maker to explore the data warehouse at different levels of detail, from aggregated to detailed measures. Typical OLAP operators are *Slice* (selection of a part of the dataset), *Dice* (elimination of a dimension), *RollUp* (move up into a dimension hierarchy) and *DrillDown* (reverse of RollUp). An example of OLAP multidimensional analysis carrying on a fact "sales" of a stores chain can be realized defining as measure "quantity" of sold products, and as dimensions "Time" (Month<Year), "location", and "product" (Item<Type). Then a multidimensional query can be: "*How many TVs have been sold per month and store?*".

Spatial OLAP (SOLAP) systems are Spatial Decision Support Systems [17] integrating Geographic Information Systems (GIS) and data warehouses and OLAP systems. Spatial OLAP extends the multidimensional model defining spatial dimensions, spatial measures and Spatial OLAP operators. An overview is presented in [9]. Spatial dimensions present levels with a spatial attribute (the geometry). Spatial measure is defined as the result of spatial operators and/or directly derived numerical values. An example of spatio-multidimensional application is shown in Figure 1-Left. It analyzes the pollution values of departments and regions according to time and pollutants. It allows answering queries such as: "What are the pollution values per month, department, and pollutant?" [10]. SOLAP tools can be classified using the taxonomy proposed by [7]: *OLAP dominant*, *GIS dominant* and *OLAP-GIS integrated*. OLAP dominant tools present all advanced OLAP functionality, but their spatial analysis and cartographic interaction/visualization capabilities are very limited. GIS dominant solutions simulate the OLAP server by providing only some basic multidimensional modeling and querying functionality. Finally, OLAP-GIS integrated solutions are most well-suited for an effective spatio-multidimensional analysis as they combine in a unique and interactive framework advanced OLAP and GIS functionality [7].

These solutions present a user interface which supports multidimensional analysis using maps, tabular, and graphical data representation in a concerted, interactive, and synchronized way (see Figure 1-Right). Synchronization means that an action (i.e. drill-down or roll-up) on one interface widget (i.e. map) has to be propagated on the other ones (i.e. pivot table and graphic displays).

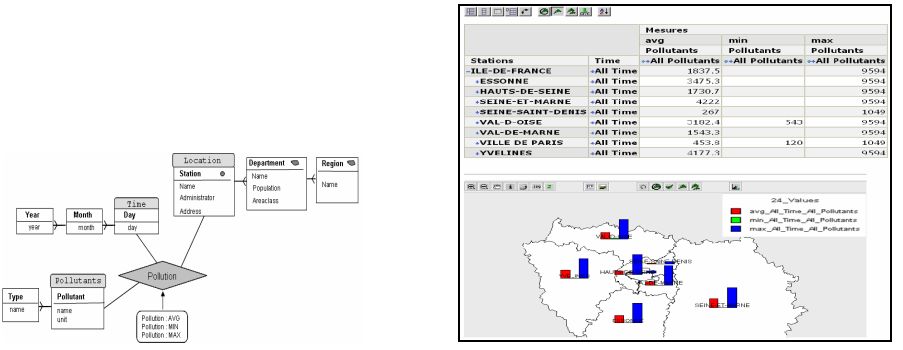


Fig. 1. A SOLAP schema (left), and a User interface of a SOLAP tool integrating OLAP and GIS functionalities (right) [10]

Figure 1-Right shows the visualization of a spatio-multidimensional query on the spatial datawarehouse of Figure 1 using the SOLAP system GeWOlap [10]. Note that spatial dimension "Location" is represented as a map while measures ("PollutionAVG", "PollutionMIN" and "PollutionMAX") are displayed using bars. Existing OLAP dominant and OLAP-GIS integrated solutions [9] try to integrate GIS analysis and cartographic visualization techniques into OLAP ones. They usually adopt 2D cartographic components and classical GIS visual encodings. Only [5] presents a SOLAP tool which integrates ERSI ArcGIS and Proclarity to visualize SOLAP queries results using 3D maps.

Existing SOLAP tools do not allow to search and add geographic and multimedia data from the Web to the spatial data warehouse. Indeed, only [6] contextualizes SOLAP data with multimedia data, but the association of photos and videos to dimensions and measures is done during the construction of the spatial data warehouse.

A geobrowser is a software that provides access to rich spatial data sets and sophisticated and intuitive interfaces through which they may be explored. Examples of geobrowers are Google Earth, Microsoft Virtual Earth, NASA WorldWind. Recent work has shown that Google Earth provides a good and extensible Geovisualization framework [12, 22]. Indeed, it allows exploring 3D geographic data, collecting geographic data from the web, supporting multimedia data on top of geographic layers, using advanced visual encoding for thematic data, and representing temporal data. Moreover, the approach presented in [23] provides "query and reasoning" spatial analysis methods [19] such as filtering by time, space and attributes, using Google Earth. They also introduce new visual encodings, and permit to aggregate data by using server-side scripts and geographic layers containing pre-aggregated data. However, none of the above mentioned projects uses data warehouse and OLAP systems, thus limiting multidimensional analysis.

To conclude, to best of our knowledge no work tries to integrate Geovisualization functionalities of geobrowsers to OLAP systems into a unique, flexible and interactive framework.

3 The GooLAP System

In this section we describe the main features offered by the Goolap system.

Main Features. The key idea underlying the design of the user interface of our system was to complement a widely-adopted tabular representation of the data with a flexible, web-based Geobrowser, providing a 3D visualization of spatial dimensions data, in order to exploit the advantages of both the textual and visual solutions. In this way a Decision Maker can perform a multidimensional exploration and analysis of information in an integrated environment, interacting seamlessly on textual and spatial representation of the data. Moreover, he/she can import different layers of spatial data from many freely available repositories, to complement the information coming from the data warehouse.

In the first implementation of this system, we adopted Google Earth as Geobrowser and JPivot as pivot table. Therefore, our system can be considered as an OLAP dominant solution, which integrates some geovisualization functionalities.

In the following we first describe these two technologies, and then present how they have been arranged together to form a coherent user interface. As for SOLAP operations, our system provides drill-down, roll-up, slice and dice operators using a pivot table and a cube navigator. Moreover it allows triggering drill operators through the simple interaction with the cartographic component of the Geobrowser.

Mondrian-JPivot. For the OLAP features, we employed two widely-adopted, freely available tools, namely Mondrian [21] and *JPivot* [16].The former is a software package designed to provide OLAP functionality in an open and extensible framework, on top of a relational database. This is achieved by means of a set of JAVA APIs, which can be used for writing applications, such as a graphical interface, for browsing the multidimensional database. These APIs can also be invoked by JSP/Servlets, within a web environment. Mondrian includes a Calculation layer, that validates and executes MDX (Multidimensional Expressions) queries, and an Aggregation layer that controls data in memory and request data that is not cached. MDX is a standard language to query multidimensional databases, just like SQL for the relational ones. In order to guarantee the greatest flexibility, to interface the relational data, an XML description of the multidimensional application has to be written.

JPivot is a software package designed for providing a web-based, graphical presentation layer on top of Mondrian. It provides specific JSP tags for easily building powerful graphical interfaces, suited to explore the data warehouse. JPivot provides functionality to modify the visualization of the pivot table and triggers the desired OLAP operators: *drill-down replace, drill-down position, expand-all,* and *drill-through*. The drill-down replace operator enables drilling from one pointed member to its child members in the dimension hierarchy, hiding the parents, whereas the drill-down position shows the parents. The expand-all operator enables drilling from all visible members in the table to child members.

Google Earth. Google Earth (shortly GE) is a virtual globe, currently freely available for personal use. It is provided in two versions: as a stand-alone application, for PC running on Windows, Mac OS, Linux and FreeBSD, and as a browser plug-in (released on June 2008) for Firefox, Internet Explorer 6 and 7. GE combines satellite

raster imagery, with vector maps and layers, in a single and integrated tool, which allows users to interactively fly in 3D from outer space to street level views. It currently incorporates data about almost every place in the world, with a typical resolution of 15 meters per pixel (although most datasets from USA and Europe are available at 1 meter resolution). A very wide set of geographical features (streets, borders, rivers airports, etc.), as well as commercial points of interest (restaurants, bars, lodging, shopping malls, fuel stations, etc…), can be overlaid onto the map. A key characteristic of this tool is the fact that the spatial datasets are not stored on client computers, but they are streamed, upon request, from Google's huge server infrastructure, ensuring fast connections and almost 100% up time. This guarantees that data are always up-to-date. Another remarkable feature implemented by GE is the ability to render a Digital Elevation Model (DEM) of the terrain, mainly using data collected by NASA's Shuttle Radar Topography Mission (SRTM). The internal coordinate system of Google Earth is geographic coordinates (latitude/longitude) on the World Geodetic System of 1984 (WGS84) datum.

From an application development point of view, GE offers two complementary ways of interaction: an ad-hoc file format, to present spatial features, and a set of APIs. More specifically, GE supports an XML grammar and file format, named Keyhole Markup Language (or KML) [18], suited to model one or more spatial features to be displayed. Through KML files, a developer can assign icons and labels to a location on the planet surface, specify camera positions to define views, add basic geometrical shapes, and so on. At the same time, the GE browser Plug-in offers a set of JavaScript APIs, allowing developers to place and control GE into web pages.

In our proposal, we exploited the 3D capabilities provided by the GE browser Plug-in to combine information from the data warehouse with real world infrastructures and geographic features. Moreover, currently many repositories and communities, containing a very broad amount of informative layers are available on the Web and can be freely and effortlessly integrated in any application exploiting Google Earth.

The User Interface. The current version of the user interface is composed of two main panels, integrated into a dynamic web page: a Geobrowser on the left and a pivot table on the right.

The pivot table, representing the OLAP hypercube, is responsible for showing the textual data and providing the hierarchical navigation across the dimensions of the data warehouse.

On the other hand, the Geobrowser is responsible for rendering in 3D the spatial information over a geo-referenced satellite image, potentially enhanced by additional informative layers, such as roads, cities, points of interest and so on. In this version, the Geobrowser mainly uses histograms to show data values, but we plan to use also pie charts. Moreover, by clicking on a geographical area, the system performs a drill-down on that dimension.

In the following we present an example of multidimensional exploration and analysis of a spatial data warehouse using the Goolap interface. This description is supported by a preliminary case study that uses a simulated dataset on pollution values in various Italian regions.

Multidimensional Navigation. In order to perform a multidimensional navigation, a Decision Maker would analyze the average value of the pollution in Italy by year and by pollutants (Fig. 2). The pivot table and the Geobrowser show the same information. In particular, the latter displays the boundaries of Italy and the pollution value using a bar. Thus, all the information is aggregated.

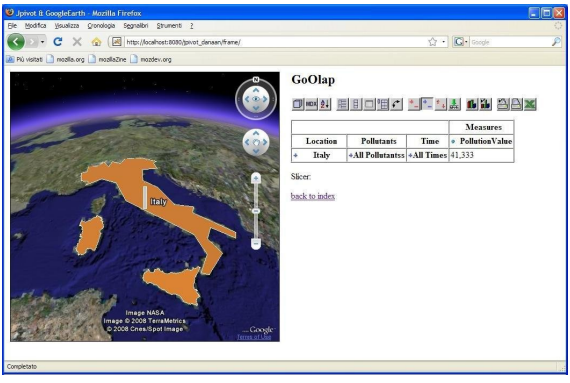


Fig. 2. Pollution value of Italy

Let us suppose that subsequently the Decision Maker is interested in knowing the pollution values for the Italian regions.

Then, by simply clicking on the nation inside the Geobrowser, he/she performs a drill-down operation of the spatial dimension "Location" moving to the "Region" level. Goolap updates the pivot table and the map at the same time, accordingly.

At this point, the pivot table shows the average pollution value for Italy and for all its regions. In the same way, the Geobrowser displays a map highlighting the boundaries of all Italian regions, and one bar is placed onto each region to visually represent the pollution value (Fig. 3). As further analysis of the data warehouse, let us suppose that the Decision Maker is interested in exploring the pollution values per type of pollutant.

Using the pivot table, he/she could apply the drill-down operator on the "Pollutants" dimension by simply clicking on the member "All Pollutants" in the Pivot table. Goolap triggers this OLAP operator and thus both the Pivot table and the Geobrowser are updated accordingly. Then, two bars, corresponding to organic and inorganic pollutants, are displayed for each Italian region (Fig. 4).

Finally, the spatial analyst could be interested in two particular regions, for instance Lazio and Campania. Using the *Cube Navigator* feature provided by jPivot, she/he can select these two regions and the visualization tool will show only the pollution values for organic and inorganic pollutants associated with those two regions(Fig. 5).

We are currently working to extend the system to support different ways to visually represent data. Indeed, Fig. 6 shows a prototype of the interface, where the pollution values are rendered using pie charts. This metaphor conveys information in a different fashion, highlighting different kinds of relationships among data.

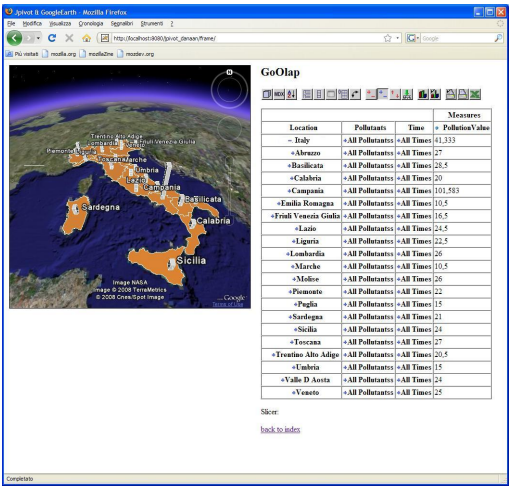


Fig. 3. Drill-Down on Italy: Pollution value per region

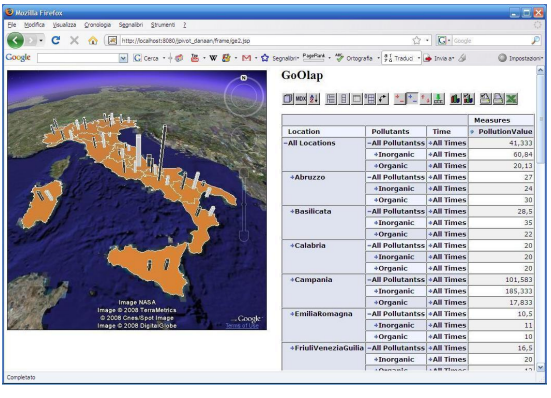


Fig. 4. Drill-down on Pollutants dimension: Pollution value per region and types of pollutants

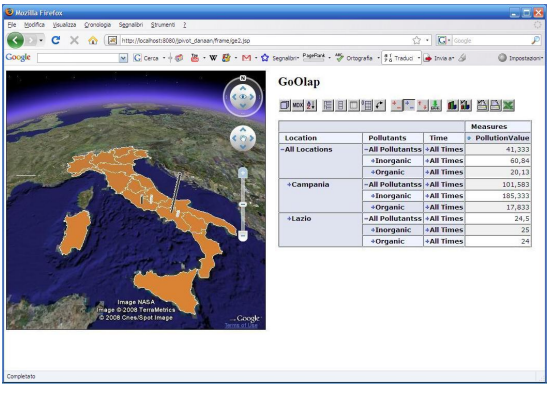


Fig. 5. Slice: pollution value for two regions

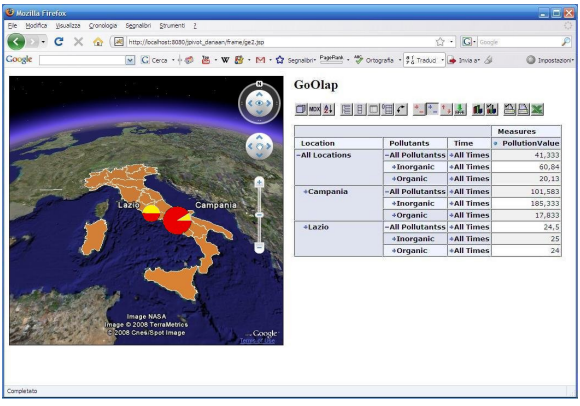


Fig. 6. Thematic map with pie charts

4 Architecture

In this section we describe the underlying architecture of the developed environment, which is responsible to merge together all the different technological solutions we have adopted. The main rationale behind the chosen solution was to obtain a flexible architecture, where each module is characterized by a loose coupling, in order to permit to replace the adopted technologies with potential novel solutions. For instance, the system could be easily modified to adopt different Geobrowsers, such as Microsoft Virtual Earth, or other pivot tables. In particular, the *Goolap* proposal relies on a three-tier architecture, composed by a data storage at the back end, a business logic layer, and two visualization components at the front end, arranged as shown in Fig. 7. To achieve modularization, all the communication among or intra modules are carried out through standard protocols and file formats, such as XML, HTML, and MDX. The Goolap architecture is structured as follows:

1. ***Visualization Tier.*** The main goal of this layer is to present both the spatial and textual information to the Decision Maker, and to notify the underlying layers with the OLAP operations he/she performed. This tier encompasses two main COTS tools: the Geobrowser, able to render user's selected information onto a map, and the Pivot table, to show textual information. They are suitably integrated through some JavaScript code, which logically forms the *Visualization Manager* module. In particular, it notifies the Logic Tier about the operation the user wants to do (e.g.: roll-up) and the data element on which to perform the operation. As a response the *Visualization Manager* receives data to refresh the Pivot table, and a KML file showing the spatial information according to the user actions.

2. ***Logic Tier.*** This layer is responsible for performing the (S)OLAP operations required by the user, and integrating data coming from the data warehouse with spatial information coming from external data stores in a single data structure, to be fed to the Visualization Layer. This tier is composed of three main modules, the *Data Manager*, the Geo *Manager* and the *Data Merger*.

a. **Data Manager.** This module interacts with Mondrian (the OLAP server) for querying and aggregating data in the Data Warehouse according to the user actions, and formats the resulting information in a XML file, described below, suitable for further processing.

b. **Geo Manager.** This module queries a database containing the geographical boundaries of the regions and states involved in the data analysis.

c. **Data Merger.** This component is responsible for both handling the communication from the *Visualization Tier*, and merging the information coming from the two other managers to feed the top layer. In particular, in this module we designed and implemented some routines for an on-the-fly generation of KML files, containing the spatial information required by the user, which are shown in the Geobrowser.

Data Manager and *Data Merger* interact through XML files. In this way they are totally decoupled. Moreover, since they are intended as interfaces, they could be implemented in various ways for different technological solutions (e.g. a different *Data Merger* to work in conjunction with *Microsoft Virtual Earth* instead of *Google Earth*). This allows us to achieve a less tight coupling between the data warehouses and the visualization tools, making them independent and replaceable.

3. ***Data Tier.*** The third layer is responsible to store and retrieve information from both the data warehouse and the database of geographical boundaries. To this aim, an OLAP server is included, *Mondrian*, interacting with a relational DBMS (in our example we adopted *Postgres*). At the same time, also the data store containing data on the geographical boundaries of regions and nations is stored in a relational database.

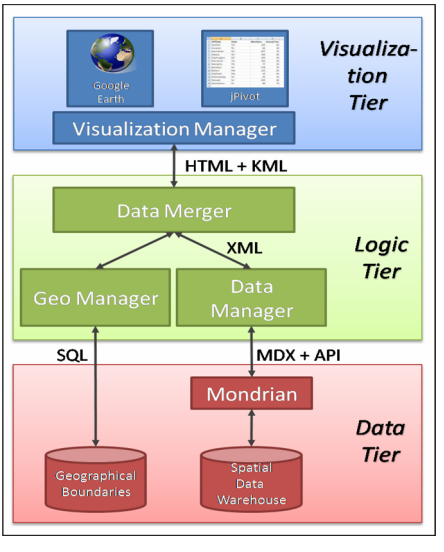


Fig. 7. Goolap Architecture

5 Conclusions and Future Work

To date there is a strong need for augmenting OLAP tools with geovisualization features, in order to provide Decision Makers with powerful instruments to get insight on hidden patterns, relations, and knowledge in the data stored in spatial data warehouses. In this work, we presented a Web-based SOLAP tool, Goolap, which integrates the OLAP system Mondrian-JPivot with Google Earth, in order to enhance OLAP analysis with the geovisualization techniques provided by the geobrowser. Therefore, our proposal presents a new SOLAP architecture which integrates OLAP and geovisualization systems. This proposal enhances existing SOLAP tools in several aspects. Goolap offers a 3D visualization of spatial dimensions data. It allows integrating information in the data warehouse with geographic information on the Web, in order to improve the geo-spatial context, which is mandatory for Spatial Decision Support Systems. Moreover, different visual encodings of the information can be used for the multidimensional analysis, to better fit the Decision Maker mental model of information. Currently the main technical open issue we are dealing with is the handling of temporal aspects in the proposed framework. At the same time, we are starting to experiment the system on a real data warehouse. Indeed, even if it is recognized that geovisualization tools should be provided to perform SOLAP operations, only an empirical usability study, with real data and a sample of Decision Makers could provide real feedbacks on the effectiveness of the solution.

Acknowledgements. The authors wish to thank Dott. Vincenza Anna Leano for her support in the development of the solution proposed in this paper.

References

1. Andrienko, N., Andrienko, G., Gatalsky, P.: Exploratory Spatio-Temporal Visualization: an Analytical Review. Journal of Visual Languages and Computing 14 (2003)
2. Andrienko, N., Andrienko, G.: Exploratory Analysis of Spatial and Temporal Data – A Systematic Approach. Springer, Heidelberg (2005)
3. Bédard, Y.: Spatial OLAP. In: Proceedings of 2nd Forum annuel sur la R-D, Géomatique VI: Un monde accessible, Montréal, November 13-14 (1997)
4. Bédard, Y., Merrett, T., Han, J.: Fundaments of Spatial Data Warehousing for Geographic Knowledge Discovery. In: Geographic Data Mining and Knowledge Discovery. Taylor & Francis, London (2001)
5. Bédard, Y., Proulx, M., Rivest, S.: Enrichissement du OLAP pour l'analyse géographique: exemples de réalisation et différentes possibilités technologiques. R. Nouvelles Tech. de l'Inf., Entrepôts de données et l'Analyse en ligne 1-20 (2005)
6. Bédard, Y., Proulx, M., Rivest, S., et al.: Merging Hypermedia GIS with Spatial On-Line Analytical Processing: Towards Hypermedia SOLAP. In: Geographic Hypermedia: Concepts and Systems. Springer, Berlin (2006)
7. Bédard, Y., Rivest, S., Proulx, M.J. (Université Laval, Canada), Spatial On-Line Analytical Processing (SOLAP): Concepts, Architectures and Solutions from a Geomatics Engineering Perspective. In: Data Warehouses and OLAP: Concepts, Architectures and Solutions, ch. 13. IRM Press, Idea Group (2007)

8. Bimonte, S., Di Martino, S., Ferrucci, F., Tchounikine, A.: Supporting Geographical Measures Through A New Visualization Metaphor In Spatial OLAP. In: ICEIS 2007, Funchal, Portugal, INSTICC (2007)
9. Bimonte, S.: On Modelling and Analysis of Geographic Multidimensional Databases. In: Data Warehousing Design and Advanced Engineering Applications: Methods for Complex Construction, 2008. Idea Group Publishing (2008)
10. Bimonte, S., Tchounikine, A., Miquel, M.: Spatial OLAP: Open Issues and a Web Based Prototype. In: 10th AGILE International Conference on Geographic Information Science, Aalborg, Denmark, May 8-11 (2007)
11. Bleisch, S., Nebiker, S.: Connected 2D and 3D visualizations for the interactive exploration of spatial information. In: Proc. of 21th ISPRS Congress, Beijing, China (2008)
12. Compieta, P., Di Martino, S., Bertolotto, M., Ferrucci, F., Kechadi, T.: Exploratory spatio-temporal data mining and visualization. Journal of Visual Languages and Computing 18(3), 255–262 (2007)
13. Costabile, M.F., Malerba, D. (ed.): Special Issue on Visual Data Mining. Journal of Visual Languages and Computing 14, 499–501 (2003)
14. Franklin, C.: An Introduction to Geographic Information Systems: Linking Maps to databases. Database, 13–21 (1992)
15. Inmon, W.H.: Building the Data Warehouse, 2nd edn. Wiley, New York (1996)
16. JPivot 2008. The JPivot Project web site (2008), `http://jpivot.sourceforge.net/` (last visited on December 07, 2008)
17. Keenan, P.: Using a GIS as a DSS Generator. In: Perspectives on Decision Support System. Grèce University of the Aegean, pp. 33–40 (1996)
18. KML 2008. The KML file format specifications (2008), `http://code.google.com/intl/en/apis/kml/` (last visited on 1, 07, 2008)
19. Longley, P., Goodchild, M., Maguire, D., Rhind, D.: Geographic Information Systems and Science. John Wiley & Sons, New York (2001)
20. MacEachren, A., Gahegan, M., Pike, W., Brewer, I., Cai, G., Lengerich, E., Hardisty, F.: Geovisualization for Knowledge Construction and Decision Support. IEEE Computer graphics and application 24(1), 13–17 (2004)
21. Mondrian 2008. The Mondrian Project web site (2008), `http://mondrian.pentaho.org/` (last visited on December 07, 2008)
22. Slingsby, A., Dykes, J., Wood, J., Foote, M., Blom, M.: The Visual Exploration of Insurance Data in Google Earth. In: Proceedings of Geographical Information Systems Research UK, Manchester, UK, pp. 24–32 (2007)
23. Wood, J., Dykes, J., Slingsby, A., Clarke, K.: Interactive visual exploration of a large spatio-temporal data set: reflections on a geovisualization mashup. IEEE Transactions on Visualization and Computer Graphics 13(6), 1176–1183 (2007)

An Automated Meeting Assistant: A Tangible Mixed Reality Interface for the AMIDA Automatic Content Linking Device

Jochen Ehnes

Centre for Speech Technology Research, University of Edinburgh
10 Crichton Street, Edinburgh, U.K.
jehnes@inf.ed.ac.uk

Abstract. We describe our approach to support ongoing meetings with an automated meeting assistant. The system based on the AMIDA Content Linking Device aims at providing relevant documents used in previous meetings for the ongoing meeting based on automatic speech recognition. Once the content linking device finds documents linked to a discussion about a similar subject in a previous meeting, it assumes they may be relevant for the current discussion as well. We believe that the way these documents are offered to the meeting participants is equally important as the way they are found. We developed a mixed reality, projection based user interface that lets the documents appear on the table tops in front of the meeting participants. They can hand them over to others or bring them onto the shared projection screen easily if they consider them relevant. Yet, irrelevant documents dont draw too much attention from the discussion. In this paper we describe the concept and implementation of this user interface and provide some preliminary results.

Keywords: Meeting assistants, Meeting processing, Mixed reality(MR), Projected user interface, Tangible user interface (TUI).

1 Introduction

While the main purpose of meetings is to facilitate direct communication between participants, documents play an important role in meetings as well. Documents often contain facts that are currently discussed, but they are not necessarily at hand. If these documents were available in a document management system, participants could search for them. However, participants of a meeting usually do not have the time to perform such queries often during a meeting. Therefore a system that could provide relevant documents for an ongoing discussion would be very helpful. A critical part of such a system would be the user interface. It should stay in the background as much as possible in order to not disturb the ongoing discussion by drawing too much attention to it. Yet it should be able to deliver the relevant documents to the participants as directly as possible, so they can incorporate these documents into the discussion directly with minimal effort. In this paper we describe a tangible mixed reality system as an interface for the AMIDA Content Linking Device [7], a system that suggests documents which may be of interest for an ongoing discussion. The documents suggested by this content linking

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 952–962, 2009.
© Springer-Verlag Berlin Heidelberg 2009

device were displayed on a laptop screen. Consequently a meeting participant, usually the discussion leader, has to monitor what is going on on the laptop's screen, which certainly distracts him or her from the meeting. Furthermore the laptop's display has the character of a private display. Other participants are not able to see the documents although they may be more important to them.

Of course this could be fixed easily by providing every participant with a laptop showing all the proposed documents. However, then everybody would have to check every document the system suggests. Furthermore, if a participant thinks a document is important, it still would not be straight forward to share it with the other participants. The user would have to describe the document first, so that the others can identify it among all the documents the system suggested so far. All this would lead the attention too much onto the laptop in front of the participants and away from the group. To overcome these challenges we propose to use a user interface projected on the tabletop in front of the participants. By using this less private form of display the documents suggested by the content linking device are visible to other participants as well. By furthermore providing an easy way to grab these documents and move them to other participants' places quickly, they can be moved to the participant they are most valuable to. Documents that are of interest to several participants or that are the subject of the discussion can be moved to a shared space, where they can be looked at by everybody at the same time.

2 Previous and Related Work

Since January 2004, the European AMI (Augmented Multi-party Interaction) integrated project has been building systems to enhance the way meetings are run and documented. AMI research revolves around instrumented meeting rooms which enable the collection, annotation, structuring, and browsing of multimodal meeting recordings. AMIs JFerret browser [4] allows its users to go through previous meetings to get themselves up to date if they were not able to attend these meetings. The browser can display video and audio recordings of all participants as well as the transcript of what was said by whom. Searching for keywords makes it easier to find parts of particular interest. While the possibility to look through recordings of previous meetings and being able to search for important sections by keywords is a very helpful tool, it requires direct action by the user. Furthermore, as the user has to interact with the system on a personal computer, which draws the user's attention to it and distracts from the conversation, the browser is more useful in preparation for a meeting than during the meeting itself.

An important goal of AMI however is to support meetings while they take place. An automated meeting assistant shall find relevant information based on the current discussion and offer it to meeting participants without requiring too much attention from them. The AMIDA Content Linking Device [7] is the first demonstrator of this system. The system consists of a Document Bank Creator (DBC) that gathers documents that are of potential interest for an upcoming meeting, a Document Indexer (DI) that creates an index over the document bank created by the DBC, a Query Aggregator and a User Interface, all connected via a central Hub. During a meeting, the Query Aggregator performs document searches at regular time intervals using words and terms from the

automatic speech recognition system. It produces a list of document names, ordered by relevance, based on the search results, as well as on a persistence model, assuming that documents that come up during several searches are likely to be more relevant than others that do not. The User interface finally displays the results to the user(s). As this work is about an alternative User Interface, we can't go into more details of the content linking device and refer to [7] for that.

The idea to use the table top as an interface to computers is not new. The first system of that kind known to the authors was DigitalDesk [12,13]. Its main intention was to bring together electronic- and paper-documents.

In [10,11] a similar setup consisting of video projector and camera, I/O-Bulb as the authors call it, mounted above the table was used to create applications that are manipulated using physical objects. Applications include the simulation of holography setups using physical placeholders for optical elements such as lasers, mirrors etc. or the simulation of fluids flowing around objects standing on the tabletop. An obvious advantage of this kind of user interface is their collaborative nature as several users can manipulate different physical objects on the tabletop at the same time, instead of being restricted to a single mouse in a conventional computer setup.

While being able to see what everybody else sees is a very important factor for collaboration, it sometimes is necessary to be able to sketch something down to clear one's thoughts before presenting it to the whole group. In [3] the authors presented a system that supports the discussion of virtual prototypes by a group of designers/engineers sitting around a projection table. The crucial difference here to other 3D viewers or the applications running on the I/O-Bulbs was that the content does not occupy the whole screen space. Instead the virtual prototype would be visible on a virtual piece of paper. Apparently conventional plots of CAD drawings were still used frequently during design review meetings, as it was so easy to sketch on them with a pencil to point out flaws or suggest improvements. Furthermore, one could just work out an idea on a piece of paper before presenting it to other meeting participants. The computer systems at the time were just considered a hinderance during these meetings. In order to make them more usable for such applications, the above mentioned prototype was developed. As the 3D models were displayed on the virtual pieces of paper, they were visible to everyone. Furthermore, the paper could be moved around using a tracked puck, so that they could be brought closer and rotated to a single person to allow for a more personal use. By grabbing two points, one with the puck and one with the pen, the virtual paper could be scaled, similarly to the two finger pinch scaling gesture known from the iPhone. Using tracked pens, participants could draw lines on the objects to annotate them. Furthermore, the system allowed to connect each piece of paper to one of several tracked pairs of shutter glasses to get a full three dimensional impression of the object. But as the stereo view certainly hindered others looking at the object, it could easily be switched off again by putting the glasses down for discussions. While we do not display 3D objects in our content linking system, we use the concept of having virtual pieces of paper that can be moved around using a physical device such as the puck. The Shared Design Space [5], a system consisting of four projectors for an interactive surface on a tabletop and a projector to create an interactive wall, is of interest as it not only use video cameras to track objects for interaction. Anoto Pens, digital pens that can track a

pattern of tiny dots on the surface they are writing on, are used to control the system as well as to draw onto the virtual documents (images). As we aimed for a simple interface to view existing documents, we don't provide such a feature at the moment.

3 Setup

In order to present documents found by the content linking device to meeting participants, we planned to build a system that can project these documents on the table tops in front of the participants during meetings.

As one of our goals for the implementation of this system was to provide the additional functionality without requiring big changes to the existing meeting environment, we decided to go for projection and video tracking from above the table. While a back-projection/tracking system from below a semi transparent tabletop would have allowed to detect when objects or fingers touch the surface, having to buy new tables with transparent tops and fitting back projection systems beneath them would have been a too big a change to the existing room as it was already fitted with a lot of recording equipment used by the AMI project. New furniture might have made other changes necessary, which we wanted to avoid to keep recordings comparable. Furthermore, the space required for a back projection/tracking system below the table possibly would become a disadvantage as meeting participants would not have been able to sit at the table as comfortably as usual and projected documents may come and go unnoticed if papers or other objects are put on the desk top on top of them.

In order to provide enough space for several participants, we planed to use multiple projection/tracking systems. To start we designed our system to support two users and we used one computer (Mac mini), projector (Optoma EP709) and camera (Imaging-Source DFK 21BF04) for each of them. Figure 1(a) shows our setup mounted around the projector for the presentation screen. We also included the presentation screen as a shared space into our system, by feeding this projector with a laptop computer (Mac-Book Pro).

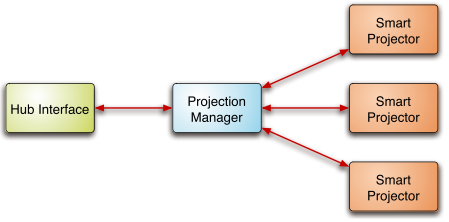
However, the software is designed to be scalable, so that we would be able to change the number of projection systems easily. While multiple projectors are often used to create a larger, tiled display [8, 6], that approach did not appear to be so suitable for our application. Desks in meeting rooms often are not arranged to create one large surface, but in different shapes, such as a "U" shape, to allow everybody to see everyone else as well as a projection screen. Some tables may not even be connected to others at all. In consequence it is not important that the projection systems form a consistent display area, as long as the user interface is consistent across the systems and documents can be moved between them in a way that is consistent to the way they are moved around on one system. Furthermore, this approach gave us the flexibility to provide individual display modes for each user, such as the shared display layer described in section 5.3.

4 System Architecture

In order to keep the number of projection systems scalable, we divided our projection system into two parts, Smart Projector an application running on all projection units and



(a) The two projection systems Mounted around the existing projector.



(b) System architecture.

Fig. 1. Prototype setup and System Architecture

a central Projection Manager. This display system is connected to the Content Linking Device via a third application, the Hub Manager (figure 1(b)).

4.1 Smart Projector

Smart Projector is the application running on every projection unit that creates the actual user interface for meeting participants. While It has a simple user interface to connect to / disconnect from the Projection Manager and to configure and activate the capturing of life video from the camera, it is switched to Fullscreen mode during normal operation. If a video input stream is available, the application searches for AR-ToolKit+ markers in it and sends the information to the Projection Manager and the applications associated with the tracked object. If no video stream is available, it displays shared documents (section 5.3) only. Additionally it also captures key board events and forwards them accordingly (section 5.2).

4.2 Projection Manager

At the center of this projection system is Projection Manager, a server that manages all important parameters of the projection units and coordinates their actions. As the Mac minis only have one display connector feeding the projector, most parameters of Smart Projector (background color, projection parameters, calibration between projection and camera coordinates, ...) are adjusted in the Projection Manager. This also makes it easy to adjust parameters for several projection units at once.

Besides managing the projection units, Projection Manager is also used to define and print the interaction devices carrying AR-ToolKit+ markers (see figure 2(b) for example) and to manage the display applications running on the system.

4.3 Display Applications

In order to keep the system easily extensible to new types of documents, we developed an API based on two base classes that can be extended to create different display

applications. A peculiarity of that API is that it consists of two base classes, one for application objects and one for display objects. The display objects are basically stateless objects that render the content they are sent by the application object on table top in the form of a sheet of paper. They are also responsible to forward certain events to the application object. The application objects on the other hand are responsible to maintain the state and change it according to user input. Whenever relevant parameters change, the application objects have to send updates to their display objects. This separation of state and display allows for an easy duplication of the display. When a display application is started on one system and is not running anywhere else, a display object as well as an application object is created locally on the projection unit. On the other hand, if the application is already running somewhere else, for example if it has been moved to the shared space (section 5.3) and consequently it has to be started on other machines, only a display object is created. This display object is then connected to the application object on the projection unit where the application was started first. After that the application object receives a call to update its display objects, so that the newly created display object displays the correct data.

4.4 Hub Interface

To connect the display system to the AMI Content Linking Device the Application Hub-Interface was developed. On one side it connects to the Hub as a consumer via the Java Native Interface, and on the other side it connects to the Projection Manager. Once the query aggregator stores new related documents in the Hub (please refer to [7] for a detailed description of the components of the content linking device), the application receives a message from the hub containing the documents name.

Upon receiving this message, the Hub Interface introduces these documents, which were all converted to PDF, into the projection system. It does so by adding a new PDF-Reader application to the list of applications and setting the Document URL as well as the Application ID to the URL of the PDF file.

Then the application is started on a projection unit specified by a popup menu. In a future version the content linking system should also provide a person or role for whom the document is most relevant. Then the documents can be sent to the best fitting person automatically. If the document is already being displayed, it is not introduced a second time. Instead the user is notified that the document could be relevant for the current discussion by bringing it to the front and letting it vibrate a little bit to create a visual ping (section 5.4).

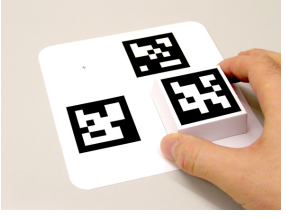
5 User Interface

In order to make the interaction with the system as direct as possible, we aimed to make the projected objects graspable. We decided against hand tracking as it is difficult to distinguish between gestures meant to manipulate documents and gesturing during discussions. This is especially true as the current setup does not allow to detect if the user's hands touch the desk top. Instead we track physical objects that serve as interaction devices using the AR-ToolKit+ tracking library.

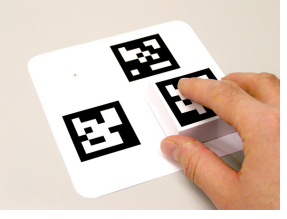
5.1 Document Handling

In order to move projected documents around, a physical object (paper grabber) is associated with them. As long as this connection exists, the virtual document follows the grabber.

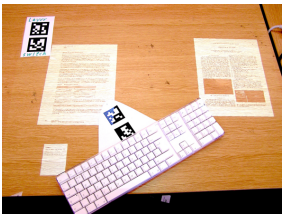
The grabber objects consist of a piece of cardboard containing three markers, one of them elevated on a box (see figures 2(b) and 2(a)).



(a) Not grabbing.



(b) Grabbing.



(c) Wireless keyboard.

Fig. 2. Input devices

The elevated marker has the functionality of a switch. By blocking the visibility of this marker using a finger for example, one tells the system to grab the document below the device. If grabbed only on the sides, so that the marker on top is fully visible, the marker is disconnected and can be moved freely. Once the marker is placed on a virtual piece of paper, users can grab the document by holding the box like a mouse and thereby covering the top marker. Because the switch marker is on the box that user's grab, they do not have to think consciously about covering the marker or not. They just have to remember the two ways of holding the grabber device; On the sides to lift the grabber from the paper or with the hand on top of it, pressing it onto the paper they want to move. Once grabbed, the document stays connected to the grabber until it is released again, i.e. the top marker gets recognized again. This may be on the same or another user's projection system.

5.2 Keyboard Forwarding

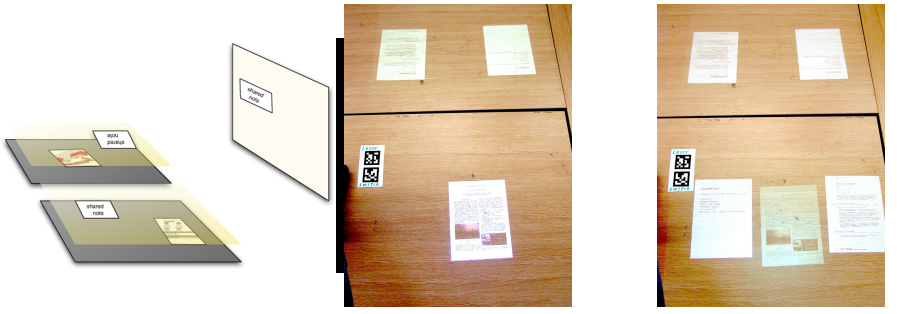
Instead of providing virtual, projected keyboards as it is usually done with touch screen interfaces, we chose to use standard wireless keyboards. In order to allow keyboard based input, a Keyboard identified by the two markers attached to it (see figure 2(c)), can be placed on a displayed document. This allows to route keyboard events to the display applications (section 4.3) that create the graphical representation of the documents. It replaces the physical connection (which keyboard is connected to which projection unit) with a virtual connection between keyboards and documents.

5.3 Sharing

In addition to augmenting the table, we wanted the system to incorporate the whiteboard as well. This way, participants are able to interact with content on the whiteboard directly from their place and move content between their space and the shared whiteboard space

easily. While Hyperdraging as described in [9] would allow participants to do that in principle, it relies on a laptop with a conventional interaction device such as a touchpad. Using hyperdraging therefore would work against our goal to let the computer disappear.

We believe it is better to 'bring the shared screen to the participant' on the press of a button, or in our case when a marker is covered by the user. We therefore implemented a shared semitransparent layer (see figure 3(a)) on top of the normal projection area which can be activated and deactivated by covering a marker placed on the projection area for that purpose. The presentation screen is the only exception here, as it does not have a private layer. It always displays the shared layer. Documents can be moved between the private and shared layers by grabbing them on one layer before switching to the other one. Once on the shared layer, all state changes such as position, orientation or which page of a multi page document is shown are forwarded immediately to other systems displaying the shared layer.



(a) Shared layer concept. (b) Shared layer off. (c) Shared layer on.

Fig. 3. Shared space

5.4 Auto-arrangement and Auto-iconizing

The Content Linking Device brings up new documents in regular intervals. In fact it often finds several documents to be displayed at the same time. In this situation it obviously is not enough to make the documents appear at a fixed location (eg. the center of the tabletop. Of course the space on the table is not unlimited, so a method had to be developed to prioritize documents and remove less relevant documents gradually. We implemented a system to arrange and iconize documents automatically. It behaves as follows: When a display application is started to present a document, it is appended to the array of automatically arranged applications. If the number of elements in this array is growing above a given limit (two applications in our case), the first element is removed and appended to the array of iconized applications. Additionally, a timer is set for each application added to the array of automatically arranged applications. Once the timer fires, the document gets iconized as well. This way documents that do not appear to be relevant to users are removed as well. If the number of elements in the array of iconized applications grows above its limit (ten applications in our case), the first application is removed and terminated. Whenever applications are added to or removed from these arrays, the applications are sent a new goal position and scale factor according to

the array and their position within that array. The first auto arranged application is displayed on the left side. The second (and latest) one is positioned next to it in the center of the projection, leaving the right side for documents the user places there to read.

Their scale factor is 1.0 so they are displayed in full size. The automatically iconized applications on the other hand are scaled down to 0.3 and arranged along the front edge of the table with the oldest one being displayed on the left and new ones being added to the right. When applications are sent to new positions or receive new scale factors, they dont change to these values immediately. Instead they animate towards these values over a given duration (1.5 seconds seemed best). This way it becomes obvious when the layout changes and it is easy to follow what is going on. This is very useful when an application that is already open is deemed to be relevant by the query aggregator again, as one can see the document move form its previous position to the position of the newest document (center). If the user places a paper grabber or keyboard connector on top of a virtual paper, it prevents the paper from being affected by the auto arrangement/iconizing system. If placed on an iconized paper, the paper is also scaled up to full size again. Now the user may move the document to where it can be read conveniently without interference of the auto arrangement system. Once the user removes the paper handling device from the projected document and no keyboard is connected with it, the system will take responsibility for it again and iconize it after 30 seconds to clean up the tabletop. For the case that the document the query aggregator determined as relevant is already displayed as the latest document or controlled by the user, a visual ping has been implemented. If pinged, a document to visually vibrates for a short period of time. It is animated by scaling it slightly up and down from its original size using a sine function. The amplitude of this vibration is scaled down to zero within 1.5 seconds to fade out the effect smoothly.

6 Results

We developed a scalable projection system to be used in meeting environments. The way it is set up allows for easy installation in existing environments. After all, the camera, projector and computer can be integrated into a single unit mounted above the tables. We implemented software components that allow for easy management and coordination of the projection units as well as a user interface based on tracked interaction devices. We demonstrated that it is easily possible to move documents around on one, as well as between different projection units or between private and shared spaces. Furthermore, the system is able to connect to the central Hub of the AMI project. This way it can be used to display documents the content linking device deems relevant for the ongoing discussion. Additional functionality to manage the displayed content automatically was implemented to cope with the stream of new documents being introduced by the content linking device.

As it turned out, the first version of the content linking device we used in our prototype had not been tuned well enough yet. During the trial meeting[1] it repeatedly brought up two agendas but rarely anything specific.

[1] Meeting ES2008d of the AMI Meeting Corpus [2]. For testing purposes meeting ES2008d is played back as input to the Content Linking Device, which searches for relevant documents from the meeings ES2008a/b/c. For more details please refer to [7].

Another weakness of the current system is the resolution of its projectors (1024x768). While this is ok for documents containing little text in a large font, such a meeting agenda or Power Point slides, it is not sufficient to display regular text documents as a virtual sheet of A4 paper.

Mark Ashdown [1] proposed to use two projectors, one to cover the whole table top with a relatively low resolution and one to cover a small area with a high resolution. However, we believe that this would effectively limit the usable display area to the small area of high resolution projection and as such would have a negative effect on our system. Participants would have to move all documents onto this foveal display area to be able to read them, which would introduce additional load to the user. It would no longer be possible to just throw a glimps at a new document. Furthermore, it would also make it impossible to place a virtual paper between two participants to look at it together, as that would be in the low resolution area.

7 Future Work

In order to address the readability problem we plan to use a projector capable of projecting full HD video (1920x1080). We furthermore plan to use it in portrait mode, effectively augmenting only one half of the users' table space. This increases the resolution of the display area further and as documents are usually printed in portrait format, it should enable us to make better use of the projected pixels.

We may also look into different approaches, such as the Microsoft Surface Computers. On the one hand they still have the disadvantages of back projection systems as described in section 3. For example they cannot be integrated into an existing setup as easily as an on-projection system and maybe more importantly, they do not allow their users to put their legs beneath the table to sit comfortably for extended periods of time. On the other hand they come as a complete unit which is easy to set up and they appear promising, as they go beyond the usual touch screens and allow to recognize objects laid onto them. This feature is necessary to attach virtual documents to physical (grabber) objects, which can easily be handed around between different meeting participants, in order to move documents between different units. However, as the resolution of the current surface computers is 1024x768, we expect them to have the same limitations as our current projection system when it comes to displaying text documents with standard sized print.

Once the display quality is sufficient for text documents and the content linking device gets better tuned, we plan to include the system in the scenarios for future AMI meeting recordings. This should give us the possibility to evaluate the system in a formal way.

Acknowledgements. This work is supported by the European IST Programme Project FP6-0033812 (AMIDA) as well as the Marie Curie Intra European Fellowship (IEF) FP7-221125 (NIPUI). This paper only reflects the authors' views and funding agencies are not liable for any use that may be made of the information contained herein.

References

1. Ashdown, M., Robinson, P.: A personal projected display. In: MULTIMEDIA 2004: Proceedings of the 12th annual ACM international conference on Multimedia, pp. 932–933. ACM Press, New York (2004)
2. Carletta, J., Ashby, S., Bourban, S., Flynn, M., Guillemot, M., Hain, T., Kadlec, J., Karaiskos, V., Kraaij, W., Kronenthal, M., Lathoud, G., Lincoln, M., Lisowska, A., McCowan, I., Post, W., Reidsma, D., Wellner, P.: The ami meeting corpus: A pre-announcement. In: Renals, S., Bengio, S. (eds.) MLMI 2005. LNCS, vol. 3869, pp. 28–39. Springer, Heidelberg (2006)
3. Ehnes, J., Knöpfle, C., Unbescheiden, M.: The pen and paper paradigm supporting multiple users on the virtual table. In: Proceedings of the Virtual Reality 2001 Conference (VR 2001), p. 157. IEEE Computer Society Press, Los Alamitos (2001)
4. Fapso, M., Schwarz, P., Szöke, I., Smrz, P., Schwarz, M., Cernocký, J., Karafiát, M., Burget, L.: Search engine for information retrieval from speech records. In: Proceedings of the Third International Seminar on Computer Treatment of Slavic and East European Languages, pp. 100–101 (2006)
5. Haller, M., Brandl, P., Leithinger, D., Leitner, J., Seifried, T., Billinghurst, M.: Shared design space: Sketching ideas using digital pens and a large augmented tabletop setup. In: Pan, Z., Cheok, D.A.D., Haller, M., Lau, R., Saito, H., Liang, R. (eds.) ICAT 2006. LNCS, vol. 4282, pp. 185–196. Springer, Heidelberg (2006)
6. heyewall. Heyewall, http://www.heyewall.de/
7. Popescu-Belis, A., Boertjes, E., Kilgour, J., Poller, P., Castronovo, S., Wilson, T., Jaimes, A., Carletta, J.: The amida automatic content linking device: Just-in-time document retrieval in meetings. In: Popescu-Belis, A., Stiefelhagen, R. (eds.) MLMI 2008. LNCS, vol. 5237, pp. 272–283. Springer, Heidelberg (2008)
8. Raskar, R., Welch, G., Fuchs, H.: Seamless projection overlaps using image warping and intensity blending. In: Proceedings of the Fourth International Conference on Virtual Systems and Multimedia, Gifu, Japan (November 1998)
9. Rekimoto, J., Saitoh, M.: Augmented surfaces: A spatially continuous workspace for hybrid computing environments. In: Proceedings of CHI 1999 (1999)
10. Underkoffler, J., Ishii, H.: Illuminating light: An optical design tool with a luminous-tangible interface. In: CHI, pp. 542–549 (1998)
11. Underkoffler, J., Ullmer, B., Ishii, H.: Emancipated pixels: real-world graphics in the luminous room. In: Rockwood, A. (ed.) Proceedings of the 26th annual conference on Computer graphics and interactive techniques, pp. 385–392. ACM Press/Addison-Wesley Publishing Co. (1999)
12. Wellner, P.: The digitaldesk calculator: Tangible manipulation on a desk top display. In: Proc. ACM SIGGRAPH Symposium on User Interface Software and Technology, pp. 107–115 (1991)
13. Wellner, P.: Interacting with paper on the DigitalDesk. Communications of the ACM 36(7), 86–97 (1993)

Investigation of Error in 2D Vibrotactile Position Cues with Respect to Visual and Haptic Display Properties: A Radial Expansion Model for Improved Cuing

Nicholas G. Lipari, Christoph W. Borst, and Vijay B. Baiyya

Center for Advanced Computer Studies, University of Louisiana at Lafayette
301 E. Lewis St., Lafayette, LA 70503, U.S.A.
ngl3747,cborst,vbb4359@cacs.louisiana.edu
http://cacs.louisiana.edu

Abstract. We present a human factors experiment aimed at investigating certain systematic errors in locating position cues on a rectangular array of vibrating motors. Such a task is representative of haptic signals providing supplementary information in a collaborative or guided exploration of some dataset. In this context, both the visual size and presence of correct answer reinforcement may be subject to change. Consequently, we considered the effects of these variables on position identification. We also investigated five types of stimulus points based on the stimulus' position relative to adjacent motors. As visual size increases, it initially demonstrates the dominant effect on error magnitude, then correct answer feedback plays a role in larger sizes. Radial error, roughly the radial difference in the stimulus and response position, modeled the systematic error. We applied a quadratic fit and estimated a calibration procedure within a 2-fold cross validation.

Keywords: Haptics, Human factors, User-calibration, Collaboration.

1 Introduction

An ongoing trend in the interaction and visualization literature is the call for increased availability of multi-sensory feedback. We present an experiment aimed at investigating Subjects' accuracy when locating a haptic stimulus presented at the palm of the hand. This study is a new work to verify a claim made in [1] concerning some systematic errors thought to be present in the experimental results. Specifically, the mean error was near zero at center of the display device, a rectangular array of vibratory motors seen in Figure 1. Error then increased as stimuli moved away from the device's center. A metric we termed radial error models the radial expansion that depends on the stimulus' radial distance from device center. The Subjects' responses are then used to create a calibration model of stimulus radius versus radial error.

Two pathways to consider when investigating systematic errors in locating position cues are physiological and psychological. In the former, mechano-receptors in the palm receiving stimuli and the physical properties of our haptic display device are a concern. For this reason, we choose stimulus points of a sufficient density, given the device's

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 963–974, 2009.
© Springer-Verlag Berlin Heidelberg 2009

resolution. Psychological operations such as the mental mapping from stimulus, to visual field, to triggering a kinesthetic response are also relevant. Here, the visual display properties of visual size and correct answer reinforcement are of interest.

There have been several recent attempts to commercialize standard input devices with integrated haptic feedback mechanisms. The VTPlayer mouse consists of two finger-pad pin arrays. Marketed toward visually-impaired computer users, recent literature [2] has shown this device to effectively convey basic directional information. In their study, both static and animated indicators were rendered for the Subjects. Other commercial devices with larger markets are Logitech's iFeel Mouse and RumblePad, the Novint Falcon, and Sensable's line of Phantom force-feedback styli.

Several works, e.g. [3,4,5,6], evaluated the effectiveness of haptic devices and the display properties relevant to efficient communication. In [3], tactors stimulated opposing sides of the wrist. The authors estimated an Information Transfer measure of 1.99 bits (almost 4 locations) for both sides combined. The experiment in [6] used a force-feedback pen device. The results suggested that 2.8 levels of stiffness and 2.9 levels of force magnitude can be perceived by Subjects. The authors of [4,5] examined subject localization ability with regard to the forearm and the abdomen. They found stimulus position relative to body landmarks to be a major factor in localization.

Several studies [7,8,9] have investigated tactile stimulus for communication of directional cues. The experiment in [8] extends the "Sensory Saltation" work of [10] to a 3×3 chair-mounted array. Eight possible directions are rendered to the Subject's back with two variations on thickness. Analysis showed responses were high above the 12.5% chance level, but recognition of thickness was negligible. A torso-mounted display in [7] provided aircraft pilots with orientation information. The authors performed simulated and in-flight tests. Within thirty minutes, Subjects correctly identified orientation by five degrees of pitch and roll.

In our original experiment [1], we addressed three parameters of a vibration pattern: position, direction, and profile. Each had one of two possible shapes: point and line. While Subjects in [1] identified each parameter at varying stages of the experiment, the focus of our newer study is position cuing. After observing mean error near zero at center of the array, then increase as stimuli moved away from the center, we postulated



Fig. 1. The vibrotactile array used in our experiment. Six rows of five pager motors are mounted on a project box containing a controller. The array is commanded via serial communication with a host computer.

some systematic error was present in the data. Another form of systematic error is discussed in [11] and concerns the position identification of a glyph's profile. That is, the vibrotactile array rendered a line with non-uniform intensity. Subjects indicated the pattern's center. It was found that responses tended to undershoot the target by a significant amount.

Envisioned for a collaborative visualization task, our device would convey supplementary information regarding other users and points of interest. Such a collaborative or guided exploration may require visual size to be scaled or correct answer reinforcement to be unavailable. These factors, along with a classification of Point Type, are examined for a position identification task. We then introduce a model of radial expansion and evaluate a proposed calibration method. This concept might then be extended to other haptic display devices, with a view to understanding the mechanisms behind this effect.

The remainder of this document proceeds as follows with a description of our haptic rendering system and the experimental methodology used in Section 2. Next, Section 3 presents the relevant statistical analysis and the calibration results. A discussion of the experimental results is then given in Section 4.

2 Experimental Methods

We conducted a repeated measures experiment to investigate the apparent trends in error and their relation to visual and haptic properties. Three Within-Groups variables were of interest: Subject (6 levels), Visual Size (3 levels), and Reinforcement (2 levels). We also considered one Between-Groups variable, Point Type (5 levels). We hypothesized that perceived position would be altered to the extent that some error would be systematic enough to be modeled for calibration.

2.1 Participants

We considered the six Subjects, S_1 to S_6, (all male) expert users for the purposes of this experiment. The use of experts reduced the effects of learning and better represented a regular user than a first time user. Although there were varying levels of prior experience with the device, each participated in previous experiment(s) with the palm-array and had training prior to data collection. Subjects S_4 and S_5 were left-handed and the rest were right-handed. Subjects S_3 and S_6 were authors of this paper and had the most experience with our display device. The median age of subjects was 26 years, with a minimum age of 24 years and a maximum age of 37 years.

2.2 Materials

Subjects in our experiment were presented with tactile stimuli from the device shown in Figure 1. We chose a set of stimulus points that well-covered the array's motors. Also, we classified each stimulus point based on its position relative to nearby motors.

Apparatus. The palm-sized vibrotactile array developed in [12] delivered stimuli during our experiment. The array consisted of six rows of five DC motors, each having a 14 mm diameter. We affixed nylon washers and foam pads above and below motors to isolate vibrations and allow a flexible-fit with the palm. A controller board housed within the project box provided serial communication with the host computer. To realize variations in intensity of vibrations, a pulse width modulation scheme was implemented on the host computer.

Software Treatment. We defined a six by five grid with each cell centered on a motor. With centers of adjacent motors separated by 18 mm, this translated into grid cells measuring 18 mm x 18 mm. Two inherent limitations of our device were a low spatial resolution of motors and significant non-zero voltage needed for motor response. We approached the low resolution through unweighted area sampling, as in [13]. Gamma correction, another technique common in graphical rendering, addressed the motor response irregularities. Our extended gamma correction equation was

$$G(x;\alpha,\mu,\gamma) = \alpha\left[(1-\mu)x^{\frac{1}{\gamma}} + \mu\right], \tag{1}$$

where x is the input stimulus magnitude, α is a scaling factor, μ sets the lowest meaningful motor voltage, and γ is the standard gamma correction parameter. More information on this method is found in [13]. Through pilot studies with Subjects S_1 and S_2, we chose the parameters $\alpha = 0.8$, $\mu = 0.25$, and $\gamma = 1.925$ to ensure the different parts of the array had similar perceived intensities.

2.3 Design

For this experiment, we considered three independent variables: Point Type (Between-Groups), Visual Size (Within-Groups), and Reinforcement (Within-Groups). Subjects will also be treated as a Within-Groups variable in analysis. The four variants of a rendered point overlapping grid cells (as seen in Figure 2) identified the special cases for the Point Type. First, C1M denoted a point rendered on the center of one motor. The second and third special cases, E2H and E2V, indicated a point between two horizontally or vertically adjacent motors, respectively. Specifically, E2H and E2V points were equidistant from two motor centers. The fourth case, termed E4N, was equidistant to four neighboring motors.

In total, there were 99 special case points, 30 C1Ms, 24 E2Hs, 25 E2Vs, and 20 E4Ns. We randomly generated 63 additional points spanning the region of interest. We called this set of points RAN. The experiment presented random permutations (randomization without replacement) of these 162 points to the Subjects in each session.

The next variable altered the rectangle in which Subjects marked answers on a graphical interface. The Visual Size had three levels: half-size (VSH), unit-size (VSU), and double-size (VSD). In the VSU level, the visual size of the rectangle in Figure 3 matched the array's size.

The presence or absence of reinforcement also varied, for the levels With Reinforcement (WR) and Without Reinforcement (WOR). During the WR level, the correct stimulus position appeared in the visual rectangle after the response was submitted.

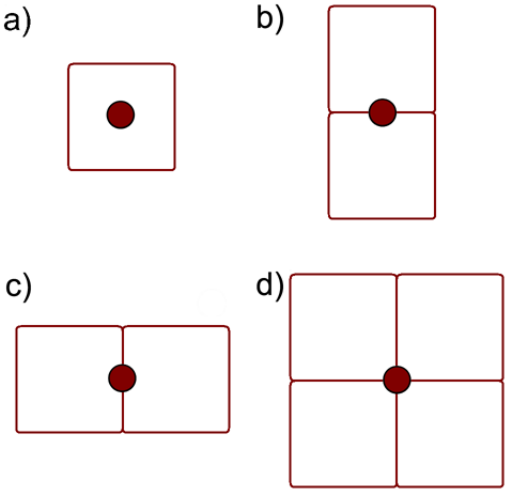


Fig. 2. Special Cases of Point Type. Squares represent motors and red circles represent stimuli. a) C1M: the center of one motor. b) E2V: equidistant between two vertically adjacent motors. c) E2H: equidistant between two horizontally adjacent motors. d) E4N: between four neighboring motors.

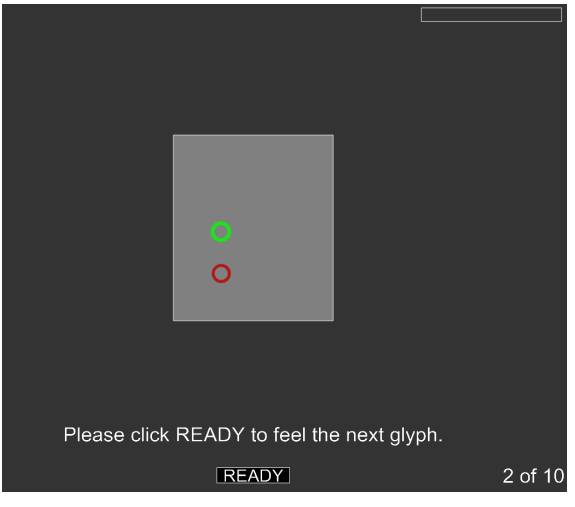


Fig. 3. A screen capture of the data collection software. Stimulus (red) and response (green) marker circles were invariant of Visual Size. A sliding timer (top-right) was active during the stimulus, and a counter (bottom-right) informed Subjects of their progress during the trial.

2.4 Procedure

We conducted an open response experiment to investigate accuracy in locating vibrotactile position cues. Subjects wore liquid-filled, noise canceling headphones with 29 dB attenuation of external sound. Each Subject completed six sessions, one per day

on non-consecutive days. Each session consisted of a Demonstration, Training, and Testing Stage. The Demonstration Stage served to illustrate the form of stimuli and to allow comfortable placement of the Subject's palm. Subjects placed their left hand on the palm-array and felt a series of short point vibrations (each lasting two seconds). The array rendered five such points and the Subjects did *not* indicate position. During Training, Subjects marked the position of ten random points at the current day's experimental condition. The Testing Stage followed. We presented the entire point set consisting of 162 distinct stimulus points to the Subject. Subjects rested for at least 30 seconds at the mid-point of testing. Sessions generally lasted 30-40 minutes. Over the six days of testing, our group of six Subjects encountered $6 \times 6 = 36$ unique permutations of the point set. The organization of sessions is discussed below.

In the Training and Testing Stages, Subjects marked the position in a rectangle rendered on a computer monitor, as shown in Figure 3. The Training allowed Subjects to become accustomed to the current conditions. Responses were recorded with a custom software package and saved in an XML file.

Organization of Sessions. The order of conditions (3 Visual Sizes $\times$ 2 Reinforcement Levels) was randomized but adhered to the following rules. On any given day, we presented all six condition combinations, one per Subject. A Subject's conditions consisted of visual levels in one order over the first three days, then the reverse order over days four through six. Reinforcement levels alternated between successive days of testing; half the Subjects started with reinforcement, half without. At least one day separated successive testing sessions of a Subject.

3 Results and Discussion

3.1 Error Magnitude Metric

We performed an ANOVA for Repeated Measures over the 36 experimental conditions (6 Subjects $\times$ 3 Visual Sizes $\times$ 2 Reinforcement Levels) and the dependent variable error magnitude. The experiment also included a Between-Groups variable of Point Type.

We report significant effects for each Within-Groups factor. Most notably, the Subjects exhibited significant differences ($F(4,5) = 92.57, p < 0.001$), as did the Visual Sizes ($F(4,2) = 8.72, p < 0.001$). WR had less overall error than WOR ($F(4,1) = 8.43, p < 0.005$). Additionally, the analysis detected interactions between Visual Size and Reinforcement ($F(4,2) = 3.51, p < 0.05$).

Post-Hoc tests with Bonferroni corrections gave us comparisons of Subjects and Visual Sizes. After aggregating results, we found that S_5 performed the best, having significantly less error than all but S_1, who was a close second. Counter to this, S_6 performed worse than all other Subjects. Interestingly, S_6 had substantially more experience than all but S_3, and neither S_3 nor S_6 was the best performing user. However, S_3 did perform more consistently over different visual sizes and reinforcements than any other user.

For Visual Size, tests showed VSH to produce less error than both VSU and VSD. The interaction between Visual Size and Reinforcement is evident in the last pairing of Figure 4. Recall that VSH's error magnitude was significantly less than VSU and VSD. As Visual Size increased, it was initially the dominant effect. Then, noting that pooled

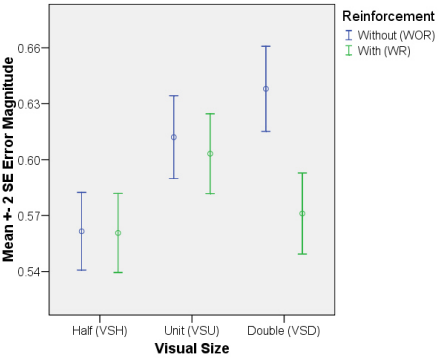


Fig. 4. Error magnitude against Visual Size and Reinforcement. The large change between the last pair of error bars illustrates the interaction between Visual Size and Reinforcement.

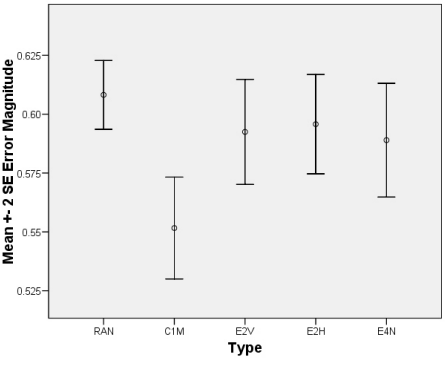


Fig. 5. Error magnitude by point type. The between-motor types RAN, E2H, E2V, and E4N were consistently higher than C1M.

VSU and VSD error magnitudes were roughly similar, Reinforcement became more of an influence at the level VSD. This observation gives credence to our initial hypothesis that both Visual Size and Reinforcement affect accuracy.

For the Between-Groups factor Point Type, Post-Hoc tests indicated a lower error magnitude at C1M points compared to RAN and E2H. E2V was a borderline result, with a near significant p-value below 0.06. This trend was consistent for all between-motor points, if not at a significant level. Figure 5 shows the near significant result of E2V as well as the similarities between other such stimuli. Further inspection of separate X and Y error components also suggests that error is smaller at motor centers than between motors. E2V was the most notable example of this.

Upon further examination of error vectors centered on stimulus points, see Figure 6, there appeared to be a trend of radial expansion, described next. As the stimulus' distance from the array's center increased, error vectors seemed to lengthen, then contract. Also, the orientation of error vectors tended to point roughly away from the array center. To measure this effect, we considered the signed metric of radial error, defined to

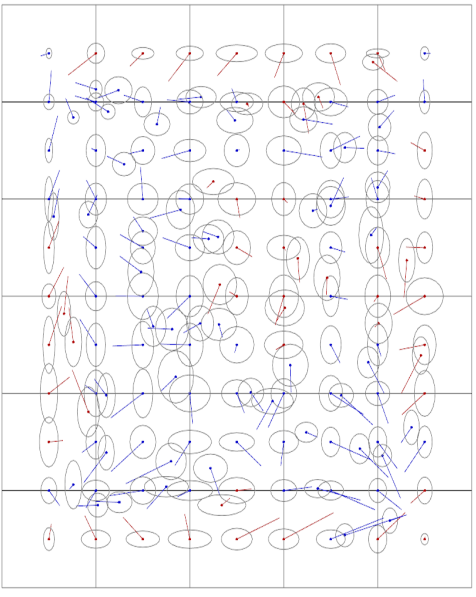


Fig. 6. Error vectors rendered at stimulus points. Ellipses represent standard error in both dimensions. Error vectors represent mean error for the point. Blue error vectors represent positive radial error, red represent negative radial error. Shown is the condition WR.

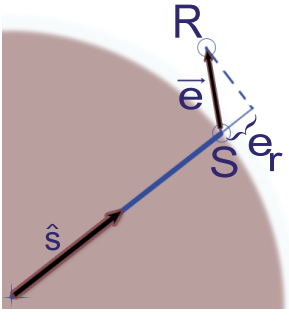


Fig. 7. Radial error derivation with the stimulus point S and associated response R. The point S defines a vector $\boldsymbol{s}$ from the origin to S, R defines the error vector $\boldsymbol{e}$ from S to R. Radial error e_r is the directed magnitude of $\boldsymbol{e}$ projected onto the normalized $\hat{s}$.

be $e_r = \hat{s} \cdot \boldsymbol{e}$, where $\hat{s}$ is the normalized radius from array center and $\boldsymbol{e} = R - S$, as in Figure 7.

3.2 Radial Error Metric

For the purposes of analyzing radial error, we transformed stimulus and response points into a canonical coordinate system such that the array center was the origin and $y' = \frac{4}{5}y$.

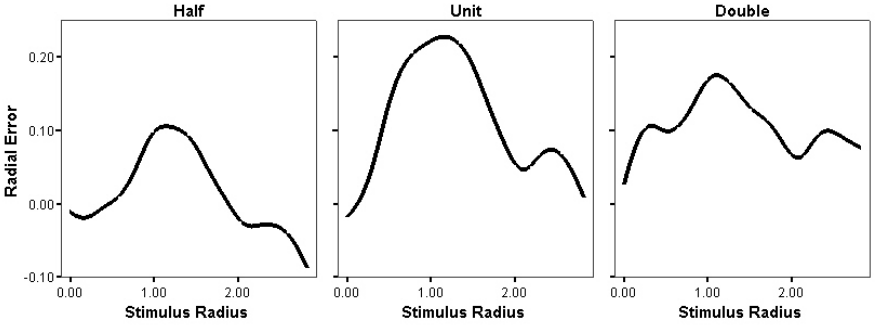


Fig. 8. Radial Error Smoothers by Visual Size. Each case contains some unimodal behavior. The Unit Visual Size (VSU) experienced the highest peak of radial error. VSH contained significantly less radial error than both VSU and VSD.

This scaling of y made all stimulus and response radii of a given length extend equally on the array. We performed another ANOVA over the metric radial error. Post Hoc tests showed VSH to cause significantly less error than VSU and VSD ($F(4,2) = 33.136, p < 0.001$). In Figure 8, however, we saw some measure of radial expansion for all three levels of the variable. VSU had a more pronounced increase, and a higher peak, than the other levels. That said, Subjects were more prone to systematic error when not given reinforcement for VSD. This can be deduced from the significant interaction between Visual Size and Reinforcement over the metric of radial error ($F(4,2) = 5.577, p < 0.05$) and the doubling of mean radial error from VSD-WR to VSD-WOR.

This radial error metric e_r gave us the directed magnitude of $\boldsymbol{e}$ projected onto $\hat{s}$, or more concisely, how far from the stimulus' radius the Subject responded. Plotting $\|\boldsymbol{s}\|$ against e_r and applying a local linear regression (smoother), we saw curves characteristic of our previous observations, e.g. Figure 8. Regression estimates started near zero at zero radius (array center) and began to rise. They then reached a clear global maximum and decreased. This suggested a quadratic relationship between stimulus radius and radial error as a preliminary model to test the feasibility of calibrating for the effect.

From the pairs $(\|\boldsymbol{s}\|, \|\boldsymbol{s}\| + e_r)$, we fitted a quadratic function f[1] via Singular Value Decomposition to a random half of the data. Then, f^{-1} estimated a stimulus radius $\|\boldsymbol{s}\|'$ that would be applied during calibration. A 2-fold cross validation (holdout method) compared the error before and after our calibration estimate as $(\|\boldsymbol{s}\| + e_r) - \|\boldsymbol{s}\|$ and $(\|\boldsymbol{s}\| + e_r) - \|\boldsymbol{s}\|'$.

Two assumptions were made concerning the inversion. The first constrained the endpoints of the model function: $f(0) \approx f(L) \approx 0$, where L was the maximum radius of the array. The second required the linear coefficient to $b < 2$. When these requirements were met, however, we were able to adjust the stimulus radius in an estimate of a calibration procedure. After calibration, we detected lower error for Subjects S_2,

[1] If the fit over $(\|\boldsymbol{s}\|, e_r)$ gives $f(x) = a + bx + cx^2$, then our fit over $(\|\boldsymbol{s}\|, \|\boldsymbol{s}\| + e_r)$ gives $f'(x) = a + bx + cx^2 + x = a + (b+1)x + cx^2$.

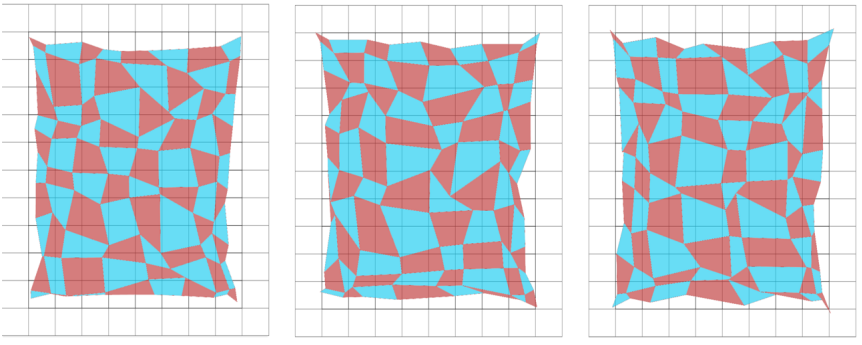


Fig. 9. Warped grid representing mean stimulus-response mapping for each level of Visual Size. Left is Half size (VSH), middle is Unit size (VSU), and right is Double size (VSD). Each colored quadrilaeral is mapped from a cell in the underlying grid. Only Point Types C1M, E2H, E2V, and E4N.

S_3, and S_5 ($F(1, 658) = 125.64, 393.92, 4.62; p < 0.05$) and for all Subjects pooled ($F(1, 3958) = 213.28, p < 0.001$). An increase in mean absolute error occurred for both S_1 and S_6, only the latter of which was significant ($F(1, 658) = 56.65, p < 0.05$). The calibration routine was not applicable to S_4's stimuli given the above constraints. Accordingly, we reported no results for this Subject.

To better visualize the effect of radial expansion over the levels of Visual Size, we constructed a warped stimulus grid from mean error vectors. As seen in Figure 9, we computed the mean response for each response point, excluding the case RAN. In each level of Visual Size, radial expansion was present to some extent. Expansion began near the center of the array, characterized by the relative area of warped to non-warped cells. The expansion also caused neighboring cells to shift from the original cell centers. A contraction near the grid edges countered the interior expansion. Mean error vectors placed edge point responses closer to the array center. These general trends reinforced our understanding of radial expansion and illustrated its relationship to Visual Size.

4 Conclusions

The statistical significance from our experimental conditions shows their importance in improving accuracy of tactile displays. When users are presented with such stimuli, care must be taken to account for different types of systematic error. The relevant factors should be identified and their effects mediated. Herein, we demonstrated such factors for positional cuing.

Our observations confirmed the research hypothesis for each condition of the experiment. Visual Size had the most effect on error magnitude for VSH and VSU, then Reinforcement had a significant effect. We observed identical results for the metric radial error and confirmed the systematic error thought to be present in a previous study. By modeling the radial error with a quadratic fit, we were able to give some insight as to the source of the error and possibly provide a calibration for it.

To varying extents, each Subject exhibited a discernable effect of radial expansion. The significant differences among users indicated another consideration for our model. Several users were clearly distinguishable from one another and others less so. This made the choice of pooled or per-user calibration difficult. The simplest effective model was a pooled, quadratic fit over all data. A significant decrease in error was achieved here. A model better fit to the data may be a quadratic regression spline with several knot points [14].

The trends above also suggested that radius alone does not fully model radial expansion. A natural choice for a second model variable would be angular measure. This would make our model a surface, multifaceted and having multiple inverses. The warped stimulus grid from Figure 9 could serve as yet another alternative model. Choices for resolution, interpolation, and regression would impact the effectiveness of such a strategy.

The effects of radial expansion and the possibility of calibrating for it were evaluated. The context, exploration of some dataset with vibrotactile position cues, informed our choice of design variables. From these concepts, extension to other tactile display devices is possible. Further studies should examine the role of Visual Size in calibration and how Visual Sizes between unit and double spread with respect to Reinforcement.

References

1. Baiyya, V.B.: Design and Evaluation of a Haptic Glyph System for 2D Vibrotactile Arrays. Master's thesis, University of Louisiana at Lafayette (2007)
2. Pietrzak, T., Pecci, I., Martin, B.: Static and Dynamic Tactile Directional Cues Experiments with VTPlayer Mouse. In: Proceedings of the Eurohaptics International conference EuroHaptics 2006, pp. 63–68 (2006)
3. Chen, H., Santos, J., Graves, M., Kim, K., Tan, H.: Tactor Localization at the Wrist, Madrid, Spain, pp. 209–218 (2008)
4. Cholewiak, R.W., Brill, C.J., Schwab, A.: Vibrotactile Localization on the Abdomen: Effects of Place and Space. Perception and Psychophysics, 970–987 (2004)
5. Cholewiak, R.W., Collins, A.A.: Vibrotactile Localization on the Arm: Effects of Place, Space and Age. Perception and Psychophysics 65, 1058–1077 (2003)
6. Cholewiak, S., Tan, H., Ebert, D.: Haptic Identification of Stiffness and Force Magnitude. In: Haptic Interfaces for Virtual Environment and Teleoperator Systems, pp. 87–91 (2008)
7. Rupert, A.H.: An Instrumentation Solution for Reducing Spatial Disorientation Mishaps. IEEE Eng. Med. Biol. 19, 71–81 (2000)
8. Tan, H., Lim, A., Traylor, R.: A Psychophysical Study of Sensory Saltation with an Open Response Paradigm. In: Proceedings of the Ninth (9th) International Symposium on Haptic Interfaces for Virtual, Orlando, FL, vol. 29, pp. 1109–1115 (2000)
9. van Erp, J.B.F., van Veen, H.A.H.C., Jansen, C., Dobbins, T.: Waypoint Navigation with a Vibrotactile Waist Belt. ACM Trans. Appl. Percept. 2, 106–117 (2005)
10. Tan, H.Z., Pentland, A.: Tactual Displays for Wearable Computing. Personal Technologies 1, 225–230 (1997)
11. Borst, C.W., Baiyya, V.B.: A 2D Haptic Glyph Method for Tactile Arrays: Design and Evaluation. In: Worldhaptics 2009 (to appear, 2009)

12. Borst, C.W., Cavanaugh, C.D.: Touchpad-Driven Haptic Communication Using a Palm-Sized Vibrotactile Array with an Open-Hardware Controller Design. In: Proceedings of the EuroHaptics 2004 Conference, pp. 344–347 (2004)
13. Borst, C.W., Asutay, A.V.: Bi-Level and Anti-Aliased Rendering Methods for a Low-Resolution 2D Vibrotactile Array. In: WHC 2005: Proceedings of the First Joint Eurohaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, Washington, DC, USA, pp. 329–335. IEEE Computer Society Press, Los Alamitos (2005)
14. Ruppert, D., Wand, M., Carroll, R.: Semiparametric Regression. Cambridge University Press, Cambridge (2003)

Developing a Model to Measure User Satisfaction and Success of Virtual Meeting Tools in an Organization

A.K.M. Najmul Islam

Information Systems Science, Turku School of Economics
Rehtorinpellonkatu 3, Turku, Finland
najmul.islam@tse.fi

Abstract. Information Systems evaluation is an important issue for the managers in an organization. But it is very difficult to evaluate. A lot of works have been done in this particular area. Many methods have been developed over the years to evaluate the information systems. The easiest and mostly used evaluation method is to measure the user satisfaction of a system. But there is no unique model that can be used to evaluate all kind of information systems. In this paper, we propose a model to measure user satisfaction of virtual meeting tools used in an organization. We verify the model by conducting two surveys and applying different statistical analysis on the collected survey data. The proposed model measures the user satisfaction and success based on six factors namely content, accuracy, ease of use, timeliness, system reliability and system speed.

Keywords: End User Computing Satisfaction, Information Systems Evaluation, Information Systems Success, User Satisfaction, Virtual Meeting Tools.

1 Introduction

The organizations are investing a large amount of money to build their Information Technology (IT) infrastructures over the years to move from the traditional paper based work to electronic means of work. They are reengineering the traditional procedures of doing their tasks by utilizing the latest development of IT products. Organizations are interested to do their works by using IT in short time, and thus gain a competitive advantage in compare to their competitors.

In 2008, the organizations continue to increase their investments on IT, even in the face of economic downturns [1]. To respond to the current economic crisis, the organizations are now bound to cut down costs in some way. Many organizations are discouraging traveling for meeting/training purposes. As an alternative to traveling the organizations are encouraging using the Virtual Meeting Tools (VMT) for meeting/training purposes. In such a situation, it has become more important to know the return of the VMT than before from organization perspectives. However, measuring the return of any IT product is always a difficult task as it is influenced by human, organizational and environmental factors [2].

Research in Information Systems (IS) evaluation has advanced with new approaches and techniques together with the continued development of the traditional

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 975–987, 2009.
© Springer-Verlag Berlin Heidelberg 2009

approaches [3]. Many review works have also been done from time to time [2], [3], [4]. Mostly in all the contributions, the user satisfaction is considered either as a construct or as a surrogate measure for the IS evaluation. But the methods to measure the user satisfaction have limitations in many cases. Neither of the existing models can be directly used to measure the user satisfaction of a VMT system. This paper tries to identify a solution to measure the user satisfaction of a VMT system, and thus measure success of the system. We refine the proposed model by doing different statistical analysis on the data collected from two surveys in the case organization.

The sequel of this paper is as follows. Section 2 describes the VMT system in general. Section 3 describes the case organization and the system where the user satisfaction will be measured. Section 4 describes how the user satisfaction is used to evaluate the IS from literature reviews. Section 5 describes the research methodology followed in this paper. In Section 6, we present the data analysis while Section 7 shows the refined model after data analysis. Finally, Section 8 draws the conclusions and describes the future research.

2 Virtual Meeting Tools in General

2.1 What Is a VMT and What Are the Offerings of a VMT

The VMT gives the opportunity to have real-time interactions using the features such as audio, video, chat tools and application sharing. The individuals use the VMT to collaborate on the group projects. Its use is observed mainly in education [22]. But nowadays, the organizations are also using this heavily for collaboration and training purposes. The learning has become more flexible by using VMT in the universities as well as in the organizations. The technology has given the opportunities to learn without being on-campus. But still, there is a sense of community even if the collaborators are thousand miles away from each other. The VMT has given the opportunity to arrange remote lectures also. Without the time and expense of travel, an expert can address a class from any location, responding to the questions of the individuals in real time.

The VMT uses common browser plug-ins and connects through a hosting service, either local or remote [22]. Most of the VMTs are platform independent, allowing users on PCs, Macs and Linux machines to share identical functionality [22]. At the appointed times, the participants log in to the system to join the meeting sessions.

2.2 Problems with the VMT

There are a number of limitations of the VMT system in general. These can affect the user satisfaction as well. The limitations are as follows.

- The VMT suffers some technical problems. For example sound and video quality can be affected by the network traffic, improper setups and other technical parameters.
- The VMT suffers social problems also. The social problems are even bigger. As with the real-time event, time zone difference is a concern. It becomes increasingly complex as the geographical range of the participants expands. For example,

connecting individuals in Canada and Mexico is straightforward, but linking the individuals in Middle East to the individuals in North America is very complicated.
- In many cases, the trainer has reduced control over the class-room in compare to the on-site training. Either the trainer cannot express his idea fully to the audiences or the audiences are not able to grab the idea fully.
- The individuals' traveling is decreasing, meaning that the traveling allowances are also decreasing. It could have a negative impact on the individuals.

3 The Case Organization and VMT System

A large IT company is our case organization. Two groups of people are used to conduct the research.

One group of people comes from a unit inside the case organization. The unit has many branches in geographically distributed locations such as America, Asia and Europe & Middle East. Many of the individuals in the unit frequently need to collaborate between branches as well as with other units inside the organization.

The individuals in the case organization are required to collaborate with the partner organizations also. The second group of people is the external collaborators from the partner organizations. These collaborators come from different organizations.

For online meeting purpose a web-based VMT is provided by the IS department of the case organization. By using the VMT, an individual can easily share his/her screen to the others, can chat with others. The studied VMT system does not support voice. However, the individuals can use phone lines or a Voice over Internet Protocol (VoIP) application to perform a conference call and talk with each other while using the VMT. In the current study, we do not intend to measure the user satisfaction of the voice conference calls rather our focus is only on the VMT. However, our case organization has recently adopted a new VMT that supports the voice conference as well. In the future, we will measure the user satisfaction of the newly adopted VMT by using the proposed model of this research.

We do not include the detailed functionality of the VMT used by the case organization here due to the confidential issues. The major steps from arranging to attending a meeting using the VMT is given in Fig. 1. From this figure, we see that the process from arranging to attending a meeting requires five main steps. The steps are detailed in the following.
- Step-1: The meeting organizer logs in to the system by using his user name and password.
- Step-2: The organizer creates a temporary account for the external partner collaborator (if any) by entering the details of the collaborator such as the user name, name, email, GSM number and time zone of the collaborator. The system automatically generates a password for the created collaborator account. If the collaborator's GSM number is provided, the user name and password are sent to his mobile phone as an SMS. Otherwise, the system shows the password and it is the duty of the organizer to remember/store the password to some place and let the collaborator know latter. This step is not necessary if there is no external collaborator required in the meeting.

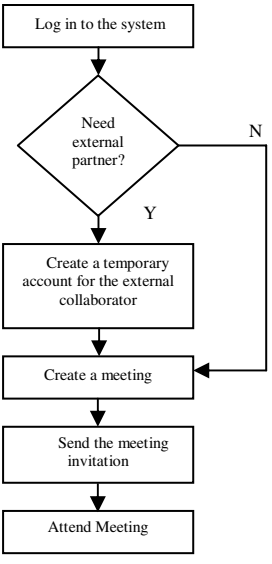


Fig. 1. Process from arranging to attending a meeting by using the studied VMT

- Step-3: The organizer creates a meeting by entering the meeting name, meeting password, schedule, participants list and other details.
- Step-4: The organizer sends invitation of the meeting to all the participants by email. The email is generated automatically by the system. The email contains the details of the meeting such as the meeting name, schedule, meeting password and a web-link to attend the meeting.
- Step-5: The participants and the organizer enter the web-link and the system asks for the user name and password. The internal participants can use their user names and passwords for logging in. However, the external partner collaborator uses the temporary user name and password created in the Step-2. After successfully logging in, a participant types the meeting name and meeting password to join the meeting.

After Step-5, the participants finally enter into the main window of the meeting tool. In the meeting tool, a participant can share his entire screen/frame of his screen/a program. A participant can allow others to control/edit his file. The tool provides chat option too. The participant list is also shown on the right side of the window. We are not allowed to show a snapshot of the tool due to the confidential issues of the case organization.

The external collaborators are not able to perform the first four steps. They are only able to attend the meeting by using the temporary user name and password created in Step-2 by the meeting organizer. We believe that the opinion about the studied VMT system of these two groups will differ. That is why; we analyze the data of these two groups separately in Section 6.

4 User Satisfaction in IS Evaluation

4.1 User Satisfaction as an Evaluation Construct

To measure the success of IS, the organizations are moving beyond the traditional financial measures, such as the return on investment [5]. For this purpose, there is a clear need to define the IS success variables. But unfortunately, the early attempts to define the IS success were ill-defined due to the complex, interdependent and multi-dimensional nature of IS success [2]. To address this problem Delone & McLean created taxonomy of IS success in 1992 [6]. They identified six variables of IS success: system quality, information quality, use, user satisfaction, individual impact and organizational impact. The model is shown in Fig. 2.

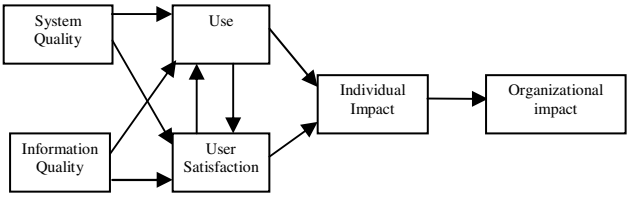


Fig. 2. DeLone & McLean IS success model

After the publication of the DeLone & McLean success model, the IS researchers started to propose different modifications on the model. For example, researchers proposed to change 'use' variable to 'usefulness' [7]. Researchers also suggested adding 'Service quality' to the model [8]. Apart from these, many other changes were proposed from different perspectives. Taking account to these changes, the DeLone & McLean model was updated by adding service quality variable [9], [10]. The updated model is shown in Fig. 3.

In both of the DeLone & McLean success models, user satisfaction was kept as an important variable to measure.

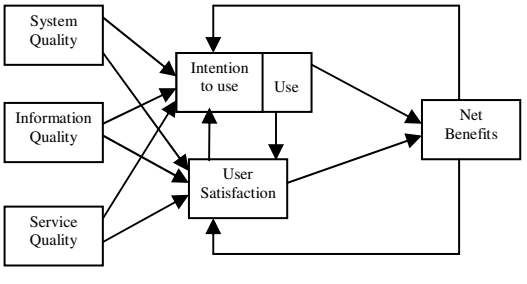


Fig. 3. Updated DeLone & McLean IS success model

4.2 User Satisfaction Measuring Models

User satisfaction has received considerable attention of researchers since the 1980s as an important surrogate measure of information systems success [11], [12], [13], [14]. Many authors believe that user satisfaction is the most useful and easiest way to evaluate the IS. Through the years, many authors have tried to develop the tools for measuring the user satisfaction. In the following paragraph we try to find the strengths and weaknesses of the most influential user satisfaction measuring tools.

Based on the review of the computer/user interface literature and the critical incident interview results, Bailey and Pearson (1983) identified 39 factors affecting the user satisfaction based on a sample of only 32 middle managers [13]. Ives *et al.* (1983) developed the User Information Satisfaction (UIS) measurement tool by the 39-item measure and a separate 4-item UIS measure to a sample of 200 production managers [11]. The emphasis was on traditional data processing environment, rather than on today's personal computing and end-user computing environment. Due to the limitations of this study, this instrument is not used as much as the End User Computing Satisfaction (EUCS) instrument developed by Doll and Torkzadeh in 1988 [15]. Doll and Torkzadeh developed a 12-item EUCS instrument by contrasting traditional data processing environment and end-user computing environment, which comprised of 5 components: content, accuracy, format, ease of use, and timeliness. The list of questions is shown in Table 1. Their instrument can be regarded as comprehensive, because they reviewed previous works on user satisfaction in their search for a comprehensive list of items. After the exploratory study was completed in 1988, many confirmatory studies with different samples were conducted, which suggested the instrument's validity [16], [17]. A test-retest of reliability of the instrument was conducted, indicating the instrument was reliable over time [18]. The instrument is widely accepted and adopted in other researches also.

Table 1. List of questions to measure user satisfaction by Doll & TorkZadeh

Identifier	Questions	Construct
C1	The system provides the precise information I need	Content
C2	The system content meets my needs	
C3	The system provides reports that seem to be just about exactly what I need	
C4	The system provides sufficient information	
A1	The system is accurate	Accuracy
A2	I am satisfied with the accuracy of the system	
F1	I think the output is presented in a useful format	Format
F2	The information is clear	
E1	The system is user friendly	Ease of use
E2	The system is easy to use	
T1	I get the information I need in time	Timeliness
T2	The system provides up-to-date information	

5 Research Methodology

5.1 Study Design

We conducted two surveys to collect the data. The sample details of the surveys are given in Table 2.

Table 2. Sample data

Location	Internal, N=112	External, N=53
Europe & Middle East	57	30
Asia	42	18
America	13	5

One survey was conducted to collect the data from the case organization. Data was collected through a random sampling in the unit. A total of 150 email invitations were sent. Out of 150, we received 112 responses those are usable in our study. So we got about 74.67% usable responses.

We conducted similar survey with the external collaborators also. A total of 80 email invitations were sent out in different partner organizations. We received 53 responses those are usable in our study. So we got 66.25% usable responses.

5.2 Instrumentation

We use the EUCS model developed by Doll and Torkzadeh (1988) in our study as the main building block. This model measures the user satisfaction based on content, accuracy format, ease of use and timeliness constructs.

In our study, we keep the five EUCS constructs and add two more constructs namely system reliability and system speed. We develop the questions related to this two extra constructs by ourselves. We add two questions (R1 and R2) related to reliability and two questions (S1 and S2) related to speed to the original EUCS model. Table 3 shows the final tool for the current study. We add two global measures of perceived overall satisfaction and success to serve as a criterion. A five scale measure was used from '5=strongly agree' to '1=strongly disagree' for all the questions.

Table 3. List of questions to measure user satisfaction

Identifier	Questions	Construct
C1	The system provides the precise information I need	Content
C2	The system content meets my needs	
C3	The system provides reports that seem to be just about exactly what I need	
C4	The system provides sufficient information	
A1	The system is accurate	Accuracy
A2	I am satisfied with the accuracy of the system	
F1	I think the output is presented in a useful format	Format
F2	The information is clear	
E1	The system is user friendly	Ease of use
E2	The system is easy to use	
T1	I get the information I need in time	Timeliness
T2	The system provides up-to-date information	
R1	I rarely see system failure	System Reliability
R2	I am satisfied with the reliability of the system	
S1	The system is loaded quickly	System Speed
S2	I am satisfied with the loading speed	
G1	I am satisfied with the overall system	Global
G2	The system is successful	

6 Data Analysis

We perform data analysis in two parts. In one part, we analyze the data collected from the individuals in the case organization. We denote this data set as 'internal data' from now on. In the other part, we analyze the data collected from the external collaborators. We denote this data set as 'external data'. We perform this separated data analysis because it is very obvious that these two groups will have different types of opinion about the system.

6.1 Factor Analysis

We conduct an exploratory factor analysis to purify the instrument. The Principal Component Analysis (PCA) is used as the extraction technique and varimax is used as the method of rotation. Some prior studies suggested using a cut-off point, 0.5 for item loadings [19]. However, Xiao and Dasgupta (2002) used a threshold value of 0.7 for factor loading criterion [20].

Table 4. Rotated factor matrix of 16-item instrument for internal data

Item	1	2	3	4	5	6	7
C1	.897						
C2	.893						
C3	.918						
C4	.767						
A1		.947					
A2		.886					
F1			.899				
F2			.876				
E1				.889			
E2				.913			
T1					.908		
T2					.719		
R1						.909	
R2						.908	
S1							.906
S2							.922

Table 5. Rotated factor matrix of 16-item instrument for external data.

Item	1	2	3	4	5	6	7
C1	.904						
C2	.863						
C3	.920						
C4	.744						
A1		.906					
A2		.920					
F1			.861				
F2			.888				
E1				.779			
E2				.860			
T1					.923		
T2					.665		
R1						.891	
R2						.827	
S1							.877
S2							.897

Table 4 shows the rotated factor matrix for the internal data. From this table, we see that the factor loadings of all the items are well above 0.7. This satisfies both of the cut-off points in the prior studies found in [19] and [20].

Table 5 shows the rotated factor matrix with the external data. From this table, we see that the factor loadings of all the items except T2 are well above 0.7 which satisfies both of the cut-off points found in [19] and [20]. However, the item loading of T2 is 0.665 which is close to the cut-off value. So, we decide to keep this item in the model at this point. We check other validity related analysis in the next.

We claim good discriminant validity with both sets of data as no cross-construct loadings above 0.5 are observed in our analysis similar to [19].

6.2 Reliability and Item-to-Total Correlation

A high Crobanch's alpha is an indication of the reliability of a construct. Table 6 shows the Crobanch's alpha of each construct. The high Crobanch's alpha in this table for all the constructs confirms the reliability of the constructs. To give an idea to the readers, we put the Crobanch's alpha of the study by Doll and Torkzadeh (1988). From the table, we find the Crobanch's alpha values of the current study are comparable with those of Doll and Torkzadeh.

Next we conduct item-to-total correlation. Following Doll and Torkzadeh's procedure, we examine the correlation score of each item with the total score of all the questions. Table 7 shows the correlation results. We choose 0.45 as the cutoff threshold. We see the item-to-total correlation is above 0.45 for all the items except F1 and

Table 6. Crobanch's alpha

Factor	Current study (internal data)	Current study (external data)	Doll & Torkzadeh
Content	0.922	0.919	0.89
Accuracy	0.896	0.876	0.91
Format	0.803	0.825	0.78
Ease of use	0.923	0.929	0.85
Timeliness	0.807	0.828	0.82
Reliability	0.949	0.903	-
Speed	0.926	0.892	-

Table 7. Item-to-total correlation

Item	Correlation Coefficient (internal data)	Correlation Coefficient (external data)
C1	.540	.569
C2	.723	.772
C3	.543	.664
C4	.695	.619
A1	.519	.451
A2	.551	.450
F1	.410	.310
F2	.391	.274
E1	.787	.839
E2	.765	.730
T1	.479	.522
T2	.627	.683
R1	.635	.612
R2	.579	.504
S1	.610	.605
S2	.637	.646

F2 (items related to format). These two items have low item-to-total correlation for both internal and external data. So, item-to-total correlation check suggests dropping F1 and F2 from the model.

6.3 Construct Validity

We perform a further analysis to check the construct validity. A correlation matrix approach is applied to examine the convergent and discriminant validity [19]. Table 8 and Table 9 show the correlation matrix for internal and external data respectively. From these tables we see that each smallest within-factor correlation is considerably higher among items intended for the same construct than among those designed to measure different constructs. This suggests adequate convergent and discriminant validity of the measurement.

Table 8. Correlation matrix of the 16-item instrument for internal data

	C1	C2	C3	C4	A1	A2	F1	F2	E1	E2	T1	T2	R1	R2	S1	S2
C1	**1.0**															
C2	**.81**	**1.0**														
C3	**.77**	**.87**	**1.0**													
C4	**.64**	**.76**	**.66**	**1.0**												
A1	.36	.36	.24	.32	**1.0**											
A2	.28	.27	.22	.06	**.81**	**1.0**										
F1	.06	.16	.17	.05	.18	.10	**1.0**									
F2	.10	.21	.13	.10	.01	.02	**.84**	**1.0**								
E1	.25	.38	.16	.44	.27	.24	.18	.20	**1.0**							
E2	.26	.38	.19	.46	.23	.20	.18	.13	**.97**	**1.0**						
T1	.04	.18	.11	.27	.28	.24	.26	.19	.36	.41	**1.0**					
T2	.30	.42	.28	.43	.21	.15	.55	.49	.52	.53	**.68**	**1.0**				
R1	.21	.49	.29	.42	.27	.25	.13	.30	.53	.45	.03	.18	**1.0**			
R2	.17	.41	.22	.37	.15	.18	.13	.30	.49	.40	.07	.21	**.91**	**1.0**		
S1	.13	.26	.18	.22	.25	.25	.19	.35	.46	.38	.21	.42	.33	.40	**1.0**	
S2	.08	.29	.26	.27	.19	.19	.13	.18	.46	.42	.33	.45	.29	.30	**.86**	**1.0**

Table 9. Correlation matrix of the 16-item instrument for external data

	C1	C2	C3	C4	A1	A2	F1	F2	E1	E2	T1	T2	R1	R2	S1	S2
C1	**1.0**															
C2	**.79**	**1.0**														
C3	**.79**	**.88**	**1.0**													
C4	**.64**	**.70**	**.69**	**1.0**												
A1	.26	.22	.18	.38	**1.0**											
A2	.13	.05	.15	.15	**.78**	**1.0**										
F1	.04	.07	.02	.21	.29	.36	**1.0**									
F2	.11	.24	.13	.09	.12	.25	**.71**	**1.0**								
E1	.40	.57	.40	.38	.21	.01	.23	.27	**1.0**							
E2	.40	.53	.43	.34	.05	.08	.29	.21	**.87**	**1.0**						
T1	.05	.21	.19	.22	.25	.18	.19	.07	.39	.41	**1.0**					
T2	.33	.48	.36	.24	.09	.08	.48	.46	.64	.63	**.71**	**1.0**				
R1	.23	.55	.38	.37	.14	.03	.14	.44	.61	.37	.02	.22	**1.0**			
R2	.30	.53	.36	.34	.03	.11	.21	.43	.58	.44	.01	.28	**.83**	**1.0**		
S1	.12	.26	.18	.07	.25	.22	.17	.38	.48	.31	.20	.46	.48	.41	**1.0**	
S2	.02	.29	.23	.08	.22	.19	.11	.19	.49	.38	.35	.46	.38	.37	**.81**	**1.0**

6.4 Criterion-Related Validity

In conducting the criterion-related validity analysis, we examine the correlation of each item with the score of the two global satisfaction criteria G1 and G2. Table 10 shows the item-to-criterion correlation. We choose the cutoff threshold of 0.4 as Doll and Torkzada's work. From the table, we see that F1, F2 and T1 (only for internal data) are below this threshold. Previously, for item-to-total correlation, we observed

low correlation coefficient value for both F1 and F2. Thus we decide to remove 'format' from our model. But we decide to keep the item, T1 in our model as it passed all other tests and the item-to-criterion correlation coefficient for external data exceeds the cutoff value. However, the item-to-criterion coefficients of F1 are very close to the cutoff value in many cases which demand further investigation. We keep it for further research.

Table 10. Item-to-criteria correlation

Item	Correlation Coefficient (internal data)	Correlation Coefficient (external data)
C1	.554	.588
C2	.694	.754
C3	.510	.551
C4	.621	.491
A1	.442	.471
A2	.459	.465
F1	.385	.395
F2	.290	.250
E1	.735	.845
E2	.706	.690
T1	.371	.405
T2	.552	.636
R1	.632	.637
R2	.596	.615
S1	.599	.576
S2	.626	.564

7 The Proposed Model and Some Discussion

Based on the data analysis results, we propose a 14-item model in Fig. 4 to measure the user satisfaction of a VMT. It consists six constructs namely, content, accuracy, ease of use, timeliness, system reliability and system speed.

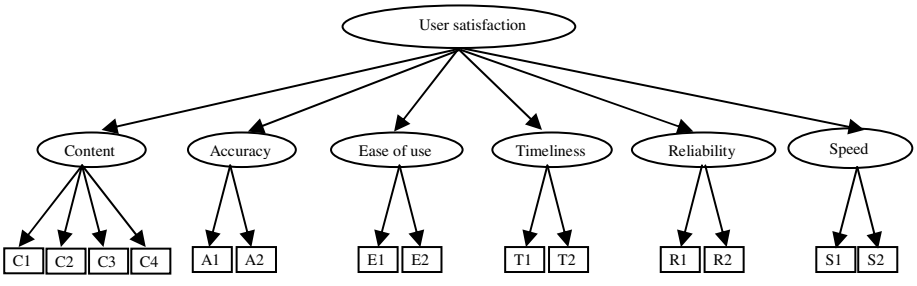


Fig. 4. The proposed user satisfaction model

Table 11 shows the item wise score of the 14-item model. From this table we get the following observations.

The internal individuals are more satisfied with the contents than the externals. This can be explained by the fact that the externals do not have enough control on the functionality of the system. For example, the externals do not have any control on the first four steps described in Section 3.

Table 11. Item wise satisfaction score

Item	Internal data		External data		Factor
	Mean	Std	Mean	Std	
C1	3.71	.64	3.58	.63	Content
C2	3.69	.63	3.54	.69	
C3	3.67	.52	3.58	.53	
C4	3.78	.53	3.69	.54	
A1	3.83	.59	3.81	.55	Accuracy
A2	3.80	.56	3.83	.54	
E1	3.81	.90	3.58	.96	Ease of use
E2	3.49	.85	3.22	.91	
T1	3.34	.67	3.05	.75	Timeliness
T2	3.51	.75	3.10	.89	
R1	3.12	.86	3.32	.67	Reliability
R2	3.05	.81	3.37	.83	
S1	4.02	.81	3.99	.87	Speed
S2	3.94	.80	3.95	.87	

Both types of individuals are equally satisfied with the accuracy of the system.

The internal individuals find the system easier to use. According to [21], we expect that the users, who mostly deal with the repetitive tasks, find the system is able to do their works properly because accessing the data becomes routine and does not involve any uncertainty over the time. It is concluded in [21] that the individuals who are more familiar with a system find the system easy to use. The internal individuals are more frequent to use the system. So they find the system easier to use than the externals.

The internal individuals find the system less reliable. Again according to [21], we can say that the more a system is used, the more its weaknesses will be identified. The internals use the system frequently and thus experience more crashes in the system. So they identify the system less reliable.

Lastly, the internal individuals are more satisfied with the speed of the system.

8 Conclusions and Future Research

We believe that the proposed user satisfaction instrument can be applied to evaluate many end user applications. We have tested this model rigorously and provided a high degree of confidence in the reliability and validity scales. However, we do not claim this model as error free. One of the main limitations of the current work is the limited number of data. We plan to test this model with other information systems by collecting more number of data in the future. Such analysis with higher number of data would provide the possibility to refine the model. For example, we have observed some contradiction regarding one format related item, which could be overcome with higher number of data. On the other hand, for many information systems, the service quality is also an important construct for measuring user satisfaction. In the future, we will try to incorporate the service questions to the model also.

References

1. Kanaracus, C.: Gartner: global IT sending growth stable. InfoWorld April 3 (2008)
2. Petter, S., Delone, W., Mclean, E.: Measuring information systems success: models, dimensions, measures, and relationships. European Journal of Information Systems 17, 236–263 (2008)

3. Smithson, S., Hirschheim, R.: Analyzing information systems evaluation: another look at the old problem. European Journal of Information Systems 7, 158–174 (1998)
4. Symons, V.J.: A review of information systems evaluation: content, context and process. European Journal of Information Systems 1(3), 205–212 (1991)
5. Rubin, H.: Into the light. CIO Magazine (2004), http://www.cio.com.au
6. Delone, W., Mclean, E.: Information systems success: the quest for the development variable. Information Systems Research 3(1), 60–95 (1992)
7. Seddon, P., Kiew, M.: A partial test and development of DeLone and McLean's model of IS success. Australian Journal of Information Systems 4(1), 90–109 (1996)
8. Pitt, L., Watson, R., Kavan, C.: Service quality: a measure of information systems effectiveness. MIS Quarterly 19(2), 173–187 (1995)
9. Delone, W., Mclean, E.: Information systems success revisited. In: Proceedings of the 35th Hawaii International Conference on Systems Sciences, p. 238. IEEE Computer Society, Hawaii (2002)
10. Delone, W., Mclean, E.: The DeLone and McLean model of information systems success: a ten year update. Journal of Management Information Systems 19(4), 9–30 (2003)
11. Ives, B., Olson, M.H., Baraoudi, J.J.: The measurement of User Information Satisfaction. Communications of the ACM 26(10), 785–793 (1983)
12. Bailey, J.E., Pearson, S.W.: Development of a Tool for Measuring and Analyzing Computer User Satisfaction. Management Science 29(5), 530–545 (1983)
13. Baraoudi, J.J., Olson, M.H., Ives, B.: An Empirical Study of the Impact of User Involvement on System Usage and Information Satisfaction. Communications of the ACM 29(3), 232–238 (1986)
14. Benson, D.H.: A Field Study of End-User Computing: Findings and Issues. MIS Quarterly 7(4), 35–45 (1983)
15. Doll, W.J., Torkzadeh, G.: The measurement of end user computing satisfaction. MIS Quarterly 12(2), 259–274 (1988)
16. Doll, W.J., Xia, W.: A confirmatory factor analysis of end user computing satisfaction instrument: A replication. Journal of End User Computing 9(2), 24–31 (1997)
17. Doll, W.J., Xia, W., Torkzadeh, G.: A confirmatory factor analysis of end user computing satisfaction instrument. MIS Quarterly 18(4), 357–369 (1994)
18. Torkzadeh, G., Doll, W.J.: Test-Retest Reliability of the End-User Satisfaction Instrument. Decision Sciences 22(1), 26–37 (1991)
19. Ong, C.S., Lai, J.Y.: Developing an instrument for measuring user satisfaction with knowledge management systems. In: Proceedings of the 37th Hawaii International Conference on Systems Sciences, pp. 1–10. IEEE Computer Society, Hawaii (2004)
20. Xiao, L., Dasgupta, S.: Measurement of user satisfaction with web-based Information Systems: An empirical study. In: Proceedings of the 8th American Conference of Information Systems, pp. 1149–1155 (2002)
21. Goodhue, D.L.: Task-System fit as a basis of User Evaluation of Information Systems: A new instrument and Empirical Test. Working paper# 93-05, MIS Research Center, University of Minnesota, Minneapolis, Minnesota (1994)
22. 7 things you should know about virtual meetings. Educause Learning Initiatives, http://connect.educause.edu/Library/ELI/7ThingsYouShouldKnowAbout/39388

Author Index

Abásolo, José 208
Abdelouahab, Zair 137
Achbany, Youssef 551, 564
Adeodato, Paulo 317
Al-Nory, Malak 363
Ali, Muhammad Intizar 172
Amghar, Youssef 501
Anacleto, Junia Coutinho 870
Andreou, Andreas S. 234
Antoniadis, Panayotis 677
Aversano, Lerina 577

Bacarin, Evandro 758
Baiyya, Vijay B. 963
Balke, Wolf-Tilo 160
Balocco, Raffaello 600
Baranauskas, Maria Cecília Calani 807, 928
Barjis, Joseph 651
Barrahmoune, Abdelaziz 196
Bastos, Ricardo Melo 376
Bauer, Mathias 745
Bendaly Hlaoui, Yousra 615
Benevides, Alessander Botti 528
Benharkat, Aïcha-Nabila 501
Bergeron, François 27
Bertolotto, Michela 940
Biajiz, Mauro 415
Bimonte, Sandro 940
Bispo, Pedro 780
Bittencourt, Ig Ibert 780
Bleistein, Steven 491
Blois Ribeiro, Marcelo 539, 627
Boehm, Matthias 40, 53
Bogdanovych, Anton 745
Bögl, Andreas 427
Bonacin, Rodrigo 807
Borst, Christoph W. 963
Boukhebouze, Mohamed 501
Breitman, Karin K. 14
Brodsky, Alexander 363

Callegari, Daniel Antonio 376
Camargo, Liadina 831
Capel, Manuel I. 479
Carrasco, A. 737
Casanova, Marco A. 14
Castellani, Stefania 819
Chang, Won-Du 918
Chen, Changqing 149
Cho, SungRan 160
Cirilo, Elder 716
Claro, Daniela Barreiro 137
Costa, Evandro 780
Cox, Karl 491
Cristal, Mauricio 627
Croteau, Anne-Marie 27
Cunha, Rodrigo 317
Cuzzocrea, Alfredo 248

De Castro, Paulo André L. 704
Dermeval, Diego 780
de Sousa Jr., José 137
Di Iorio, Angelo 90
Di Martino, Sergio 940
Dustdar, Schahram 172

Ehnes, Jochen 952
El Asri, Bouchra 196
El Morr, Christo 677
Enderlein, Sebastian 66
Escudero, J.I. 737

Ferrucci, Filomena 940
Franco, Cristiano 627
Fritzsche, Steffen 402
Fugini, Mariagrazia 445
Furtado, Antonio L. 14
Furtado, Elizabeth 831

Gad, Walaa K. 325
Garcia, Ana Cristina Bicharra 882
Ghezzi, Antonio 600
Giménez, Diego M. 639
Goederich, Marc 689
Grasso, Antonietta 819
Guizzardi, Giancarlo 528

Habich, Dirk 40, 53
Haider, Abrar 906

Helmich, Marco 66
Hernández, M.D. 737
Henning, Gabriela P. 639
Hill, Seamus 287
Hoang, Kiem 299
Hunold, Sascha 78

Ibe, Komon 491
Ihara, Masayuki 677
Islam, A.K.M. Najmul 975

Jemni Ben Ayed, Leila 615
Jerbi, Houssem 220
Joly, Adrien 677

Kamel, Mohamed S. 325
Kang, Jai W. 125
Kang, James M. 125
Kaplan, Aaron 819
Kenzi, Adil 196
Kiv, Sodany 551
Koivisto, Matti 677
Kolp, Manuel 551, 564
Krellner, Björn 78
Kriouile, Abdelaziz 196
Kroha, Petr 467
Krüger, Jens 66
Kulesza, Uirá 716

Lahiri, Tosca 790
Lamperti, Gianfranco 348
Lange, Jean-Charles 564
Lanquillon, Carsten 402
Leal, José Paulo 102
Lehner, Wolfgang 40, 53
Leme, Luiz André P. Paes 14
Lemke, Ana Paula 627
Leone, Horacio P. 639
Lima, Fernanda 858
Lima, Sinésio Teles de 858
Lipari, Nicholas G. 963
Lokman, Anitawati Mohd 894
Lopes, Denivaldo 137
Lucena, Carlos J.P. de 716

Maamar, Zakaria 501
Maciel, Cristiano 882
Madeira, Edmundo R.M. 758
Marana, Aparecido Nilceu 770
Marchetti, Carlo 90
Maret, Pierre 677
Martins, Paulo 184
Medeiros, Claudia 758
Meira, Silvio 317
Mendoza, Luis E. 479
Miani, Rafael Garcia 415
Mohebi, E. 389
Morgan, David 125
Moura, João Paulo 184
Mueller, Markus 402
Mueller, Remo 114
Müller, Jürgen 66
Mulazzani, Fabio 456

Nachev, Anatoli 287
Nagamachi, Mitsuo 894
Nash, Hadon 363
Nassar, Mahmoud 196
Neris, Vânia Paula de Almeida 928
Nguyen, Tu Anh Hoang 299
Noll, Rodrigo Perozzo 539
Noor, Nor Laila Md. 894
Nunes, Camila 716
Nunes, Ingrid 716

Oliveira, Káthia Marçal de 858
Oliveira, Rui 184
Orleans, Luís Fernando 3

Pacca, Henrique 780
Papatheocharous, Efi 234
Pedro, João 780
Penteado, Bruno Elias 770
Pereira, Vinícius Carvalho 882
Pergl, Robert 590
Pichler, Reinhard 172
Pomares, Alexandra 208
Pomberger, Gustav 427
Pramanik, Sakti 149
Preissler, Steffen 40
Prokhorov, Danil V. 265

Qian, Gang 149
Queirós, Ricardo 102

Rauber, Thomas 78
Ravat, Franck 220
Raymond, Louis 27
Reichel, Thomas 78
Renga, Filippo 600
Rink, Manuela 467

Rodrigues, Fátima 184
Rodrigues, Marcos Antônio 807
Rokach, Lior 309
Romero, M.C. 737
Roncancio, Claudia 208
Roque, Licinio 882
Roulland, Frédéric 819
Rünger, Gudula 78
Russo, Barbara 456

Santos, Marilde T.P. 415
Sap, M.N.M. 389
Schclar, Alon 309
Schilling, Albert 831
Schirinzi, Michele 90
Schrefl, Michael 427
Schwind, Michael 689
Shin, Jungpil 918
Shishkov, Boris 513
Sichman, Jaime S. 704
Siepermann, Christoph 665
Siepermann, Markus 665
Silva, Marcos Alexandre Rose 870
Silva, Marcos Tadeu 627
Simoff, Simeon 745
Sinderen, Marten van 513
Sivianes, F. 737
Soonthornphisaj, Nuanwan 275
Spahn, Michael 843
Stoyanov, Borislav 287
Struska, Zdenek 590
Subercaze, Julien 677
Succi, Giancarlo 456
Sun, Te-Hsiu 336

Teawtechadecha, Pattarawadee 275
Teste, Olivier 220
Tortorella, Maria 577
Tran, Van Anh 114
Truong, Hong Linh 172

Véras, Douglas 780
Verbraeck, Alexander 513
Verner, June 491
Villamil, María del Pilar 208
Vitali, Fabio 90

Wautelet, Yves 551, 564
Weber, Norbert 427
Willamowski, Jutta 819
Wloka, Uwe 40, 53
Woodman, Mark 790
Wulf, Volker 843

Yaguinuma, Cristiane A. 415
Yahyaoui, Hamdi 728
Yamamoto, Shuichiro 491
Yanzer Cabral, Anderson 627

Zanella, Marina 348
Zeier, Alexander 66
Zhang, Guo-Qiang 114
Zhu, Qiang 149
Zimbrão, Geraldo 3
Zurfluh, Gilles 220