Janis Grabis
Marite Kirikova (Eds.)

LNBIP 90

Perspectives in Business Informatics Research

10th International Conference, BIR 2011
Riga, Latvia, October 2011
Proceedings

Springer

Lecture Notes in Business Information Processing 90

Series Editors

Janis Grabis
Marite Kirikova (Eds.)

Perspectives in Business Informatics Research

10th International Conference, BIR 2011
Riga, Latvia, October 6-8, 2011
Proceedings

Springer

Volume Editors

Janis Grabis
Riga Technical University
Institute of Information Technology
1 Kalku, Riga
1658 Latvia
E-mail: grabis@iti.rtu.lv

Marite Kirikova
Riga Technical University
Department of Systems Theory and Design
1 Kalku, Riga
1658 Latvia
E-mail: marite.kirikova@cs.rtu.lv

ISSN 1865-1348 e-ISSN 1865-1356
ISBN 978-3-642-24510-7 e-ISBN 978-3-642-24511-4
DOI 10.1007/978-3-642-24511-4
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2011937285

ACM Computing Classification (1998): J.1, I.2.4, H.3.5, H.4

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

Business informatics combines information and communication technology (ICT) and management knowledge and skills. It is about empowering enterprises by utilizing the advantages of existing ICT solutions, creating and introducing new ICT solutions corresponding to ever-changing needs of the enterprises. These proceedings contain 25 contributed papers presented at the 10th International Conference on Perspectives in Business Informatics Research (BIR) held October 6-8, 2011, in Riga, Latvia. This year the BIR conference attracted 68 submissions from 27 countries. They were rigorously reviewed by 38 members of the Program Committee representing 18 countries. As the result, 25 full papers from 14 countries were selected for publication in this volume. The volume also includes invited papers by Dimitris Karagiannis and John Krogstie. Another 17 papers were presented at the conference and published in a separate volume of local proceedings. The papers presented at the conference cover all aspects of business informatics research, and this year there was a particular emphasis on the interplay between business process management, data management and business analytics. The BIR conference series was established 11 years ago as the result of the collaboration of colleagues from Swedish and German universities. The goal was to create a forum where researchers in business informatics, senior as well as junior, could meet and discuss their work. The conference series has a Steering Committee, to which one or two persons from each appointed organizer are invited. In its 11 years, the conference has expanded significantly, and this year a number of satellite events were organized for the first time in order to nurture the business informatics community. Four workshops as well as a Doctoral Consortium took place a day before the conference. We would like to thank everyone who contributed to the BIR 2011 conference. We thank the authors for contributing and presenting their research, we appreciate the invaluable contribution of the members of the Program Committee and external reviewers and we thank all members of the local organization team from the Riga Technical University for ensuring the smooth processing of the conference. We acknowledge the EasyChair development team for providing such a convenient tool for preparing these proceedings and the Springer publishing team for their collaboration. Special thanks to the Baltic Data company for sponsoring the Doctoral Consortium. Last but not the least we thank the Steering Committee and we hope that BIR 2011 is a valuable addition to the further development of the BIR conference series.

July 2011

Marite Kirikova
Janis Grabis

Conference Organization

Program Co-chairs

Janis Grabis	Riga Technical University, Latvia
Marite Kirikova	Riga Technical University, Latvia

Program Committee

Jan Aidemark	Växjö University, Sweden
Esma Aïmeur	University of Montreal, Canada
Eduard Babkin	State University - HSE, Nizhny Novgorod
Per Backlund	University of Skövde, Sweden
Janis Barzdins	University of Latvia
Rimantas Butleris	Kaunas University of Technology, Lithuania
Sven Carlsson	Lund University, Sweden
Peter Forbrig	University of Rostock, Germany
Horst Günther	University of Rostock, Germany
Jonas Hedman	Copenhagen Business School, Denmark
Markus Helfert	Dublin City University, Ireland
Björn Johansson	Copenhagen Business School, Denmark
Kalinka Kaloyanova	University of Sofia, Bulgaria
Dimitris Karagiannis	University of Vienna, Austria
Andrzej Kobylinski	Warsaw School of Economics, Poland
Yannis Manolopoulos	Aristotle University of Thessaloniki, Greece
Lina Nemuraite	Kaunas University of Technology, Lithuania
Jyrki Nummenmaa	University of Tampere, Finland
Jacob Nørbjerg	Copenhagen Business School, Denmark
Tomáš Pitner	Masaryk University, Czech Republic
Nava Pliskin	Ben-Gurion University of the Negev, Israel
Vaclav Repa	Prague University of Economics, Czech Republic
Hans Röck	University of Rostock, Germany
Kurt Sandkuhl	University of Rostock, Germany
Petr Sodomka	Brno University of Technology, Poland
Christian Stary	University of Linz, Austria
Janis Stirna	Stockholm University, Sweden
Eva Söderström	University of Skövde, Sweden
Bernhard Thalheim	University of Kiel, Germany
Benkt Wangler	University of Skövde, Sweden
Stanislaw Wrycza	University of Gdansk, Poland
Jelena Zdravkovic	Stockholm University, Sweden
Iryna Zolotaryeva	Kharkiv National University of Economics, Russia
Enn Õunapuu	Tallinn University of Technology, Estonia

Organizing Committee Chair

Renate Strazdina	Riga Technical University, Latvia

Steering Committee

Kurt Sandkuhl	University of Rostock (Chair), Germany
Eduard Babkin	State University - HSE, Nizhny Novgorod, Russia
Per Backlund	University of Skövde, Sweden
Rimantas Butleris	Kaunas University of Technology, Lithuania
Sven Carlsson	Lund University, Sweden
Peter Forbrig	University of Rostock, Germany
Horst Günther	University of Rostock, Germany
Marite Kirikova	Riga Technical University, Latvia
Harald Kjellin	Kristianstad University College, Sweden
Lina Nemuraite	Kaunas University of Technology, Lithuania
Jyrki Nummenmaa	University of Tampere, Finland
Hans Röck	University of Rostock, Germany
Eva Söderström	University of Skövde, Sweden
Bernd Viehweger	Humboldt University, Germany
Benkt Wangler	University of Skövde, Sweden
Stanislaw Wrycza	University of Gdansk, Poland

Additional Reviewers

Alklind Taylor, Anna-Sofia
Anohina-Naumeca, Alla
Berzisa, Solvita
Borchardt, Ulrike
Gounaris, Anastasios
Grundspenkis, Janis
Juell-Skielse, Gustaf
Kalnins, Audris
Kapocius, Kestutis
Khadraoui, Abdelaziz
Koliskins, Andrejs
Leveque, Thomas
Liepins, Renars
Ludzia, Marcin
Lyhagen, Johan
Matulevicius, Raimundas
Meyer, Martin
Murnion, Phelim
Ngosi, Theodora
Olejniczak, Jaroslaw
Pham Thi, Thanh Thoa
Plomp, Marijn G.A.
Poniszewska-Maranda, Aneta
Rencis, Edgars
Skersys, Tomas
Sprogis, Arturs
Stamer, Dirk
Stankovski, Vlado
Strazdina, Renate
Sukovskis, Uldis
Symeonidis, Panagiotis
Tsamoura, Efthymia
Vizgunov, Arseniy
Walshe, Ray
Ziema, Maris
Zikra, Iyad

Table of Contents

Keynotes

Business Information Systems Utilizing the Future Internet 1
John Krogstie

Next Generation of Modelling Platforms 19
Dimitris Karagiannis and Niksa Visic

Business Intelligence and Performance Management

Using Semantically Annotated Models for Supporting Business Process Benchmarking 29
Hans-Georg Fill

Performance Measurement Framework with Formal Indicator Definitions 44
Aivars Niedritis, Laila Niedrite, and Natalija Kozmina

Using the Entity-Attribute-Value Model for OLAP Cube Construction 59
Peter Thanisch, Tapio Niemi, Marko Niinimaki, and Jyrki Nummenmaa

Near Real-Time Data Warehousing with Multi-stage Trickle and Flip ... 73
Janis Zuters

Data and Processes

Data Consistency in Transactional Business Processes 83
Andreas Lodde, Antoine Schlechter, Pascal Bauler, and Fernand Feltz

Business Process and Regulations: Approach to Linkage and Change Management 96
Peteris Rudzajs and Ilze Buksa

Parallel Tabu Search Algorithm for Data Structure Composition 110
Eduard Babkin, Margarita Karpunina, and Natalia Aseeva

Ontologies

Survey on Ontology Languages 124
Diana Kalibatiene and Olegas Vasilecas

Advanced RDB-to-RDF/OWL Mapping Facilities in RDB2OWL 142
Guntars Būmans and Kārlis Čerāns

Export of Relational Databases to RDF Databases by Model Transformations 158
Sergejs Rikacovs

Architectures

An Analysis of Enterprise Architecture Maturity Frameworks 167
Martin Meyer, Markus Helfert, and Conor O'Brien

Enterprise Resource Planning (ERP) Systems: Use of Reference Models 178
Dejan Pajk, Mojca Indihar-Štemberger, and Andrej Kovačič

A Universal Model-Based Solution for Describing and Handling Errors 190
Sergejs Kozlovics

Stakeholders' Perspectives

Validating Organizational Knowledge Patterns: Case Study from Information Demand Modeling 204
Kurt Sandkuhl

Analysis of Dynamic Interactions with External Parties during Maintenance of ERP Systems........ 217
Igors Mikulovs and Jānis Grabis

Discovering of Users' Interests Evolution Patterns for Learning Goals Recommendation 231
Witold Abramowicz, Jacek Małyszko, and Dawid Grzegorz Węckowski

Web Information Systems and Services

Modeling Secure Navigation in Web Information Systems 239
Marianne Busch, Alexander Knapp, and Nora Koch

Quality of Health Web Sites: Dimensions for a Wide Evaluation........ 254
Álvaro Rocha, Avelino Victor, and Patrícia Leite Brandão

Analysis and Evaluation of Selected Shops with Organic Food in Poland 267
Witold Chmielarz

Evaluating the Application of Service-Oriented Auditing in the B2G Domain: A Case Study 281
Faiza Allah Bukhsh and Hans Weigand

Cloud Computing and Economic Growth in the Baltic Sea Region Countries 296
Andrzej Kobyliński and Marcin Michalski

Systems Approach

The Viable Systems Approach (VSA) for Re-interpreting Network Business Dynamics 304
Paolo Piciocchi, Clara Bassano, Erica Paduano, and Maureen Galvin

Views on Scientific Workflows 321
Mirko Sonntag, Katharina Görlach, Dimka Karastoyanova, Frank Leymann, Polina Malets, and David Schumm

Agile Business Process Management in Research Projects of Life Sciences 336
Silke Holzmüller-Laue and Bernd Göde

A Conceptual Framework for Design Science Research 345
Łukasz Ostrowski, Markus Helfert, and Fakir Hossain

Author Index 355

Business Information Systems Utilizing the Future Internet

John Krogstie

Department of Computer and Information Science
Norwegian University of Science and Technology (NTNU)
Sem Sælands Vei 7-9, 7491 Trondheim, Norway
krogstie@idi.ntnu.no

Abstract. The "Internet of Things" (IoT) has come to describe a new paradigm that enables the Internet to reach out into the real world of physical objects. Technologies like RFID, short-range wireless communication, real-time localization and sensor networks are becoming increasingly common and turning IoT into reality. It is expected to grow rapidly into a huge new market domain that may lead to disruptive changes in areas such as logistics, energy management and healthcare. Mobile and collaborative applications and services utilizing information processing and process support enabled by sensor data from a vast numbers of connected and cheap devices will change many markets when being made more easily available. New event-driven architectures (EDA) providing varied information to support collaborative decision-making enable more decisions to be made closer to the problem owner. The expected impacts of the combination of IoT and EDA on business and society are formidable. It also opens the possibility to take into account additional input from users to ensure shorter turnaround from ideas to new, personalized information systems support. Future business information systems will need to take this situation into account, addressing technological, methodological and conceptual challenges. This paper will focus on the latter, discussing in particular the potential role of model-based techniques and how to assess and improve the quality of models and modeling approaches in this setting.

Keywords: Future internet, Model-based development, IoT, BPM, AKM.

1 Introduction

ICT has over the last 30 years gone from obscure to infrastructure. All organizations are dependent on an application systems portfolio supporting its current and future tasks, and newcomers in any area are dependent on establishing new such application portfolios quickly in a way that can evolve with changed business needs, technological affordances and expectations among cooperators, competitors and customers, these days especially for the need to address a rising number of (in particular) mobile delivery channels.

Over these 30 years, one has regularly investigated how IT-systems are developed and used. In [29], results from a 1977 survey on distribution of work on IT in organizations where published. Somewhat surprisingly at that time, one found that

J. Grabis and M. Kirikova (Eds.): BIR 2011, LNBIP 90, pp. 1–18, 2011.

almost half of the time was used on maintenance (i.e. changing systems that were already in production). We have done similar investigations in Norway in 1993 [26], 1998 [14], 2003 [22], and 2008 [7]. For some areas, there are large differences. As an example, Table 1 illustrates the developments in the main programming languages being used. The use of underlying technical platform has also changed dramatically.

Table 1. Percentages of systems developed using different programming languages

	2008	**2003**	**1998**	**1993**	**Nosek/Palvia 90 [34]**	**Lientz/Swanson 77**
COBOL	4.5	0.5	32.6	49	51	52
4GL	12	13.5	16.9	24	8	
C	2.4	12.5	15.4	4	3	
C++	17.5	23.1	15.1			
C#	4.9					
RPG/ Scripts	6.7	NA	12.9	4	10	22
Java	22.6	29.8	2			
Ass.	0.4		0.9	3		12
Fortran			0.6	4	7	2.4
PASCAL			0.3	2		
PL/1			0.3	2		3.2
Other	28.9	20.2	2.6	6	21	7.7

Other areas are remarkably stable. As illustrated in table 2, the split of the time use for development vs. maintenance is still like it was 30 years ago. (In absolute terms less time is used for both development and maintenance, since most of the time used by IT-departments these days are on user-support and systems operations). Over the last 20 years the percentage of so-called new systems that are in fact replacement systems, being installed basically to replace an old system, has stayed above 50%, rising slowly.

Table 2. Comparisons of maintenance figures across investigations

Category	**2008**	**2003**	**1998**	**1993**	**Nosek/Palvia 90**	**Lientz/ Swanson 77**
Development	21	22	7	30	35	43
Maintenance	35	37	1	40	58	49
Other work	44	41	2	30	7	8
Disregarding other work						
Development	34	34	7	41	38	47
Maintenance	66	66	3	59	62	53

And even if some of the core technologies have changed as exemplified above with the transition of programming language used from COBOL to Java and other

OO-languages, many long-term trends for the development is quite stable. The following is a list of trends in ICT that was originally written more than 10 years ago:

- IT becomes infrastructure, ubiquitous and pervasive
- Increasing miniaturization
- All data potentially available from anywhere
- Increasing amount of data of increasing number of data types
- Increasing needs to structure data into information
- Increased virtualization
- Integrating analog, digital and biologically based technology

As overall trends, these have stayed and will stay (although some of the developments might slow down when Moore's low no longer applies).

One example of a notable change is where systems are developed, maintained and operate compared to 15 years ago. In 1993, 58% of the systems were developed by the IS-organization, and only one percent was developed in the user organization. In 1998, however, 27% of the systems were developed by the IS-organization and 27% as custom systems in the user organization. In 2003, 23 % of the systems are developed in the IS-organization, whereas in 2008 only 12% was developed in the IS organization. The percentage of systems developed by outside firms is higher in 2008 (40% vs. 35% in 2003, vs. 22% in 1998 vs. 12 % in 1993). The percentage of systems developed based on packages with small or large adjustments is also comparatively high (41% in 2008 vs. 39% in 2003 vs. 24% in 1998 vs. 28% in 1993). The new category we introduced in 1998, component-based development (renamed "use of external web services" in 2008) is still small (5% in 2008) although increasing (1.0 % in 2003, 0.4% in 1998). Whereas earlier the main way of developing systems where to do it in-house or using temporary consultants, outsourcing of all types of IT-developments has been on the rise over the last 10 years, although you also find examples lately on insourcing. In the 2008 investigation around a third of the IT-activity was outsourced (32.9% in private sector, 24.1 in public sector). Whereas only two of the respondents reported to have outsourced all the IT-activities, as many as 84% of the organizations had outsourced parts of their IT-activity. Whereas the public organizations have outsourced more of the development (40% in public, 29% in private) and maintenance (34% in public, 30% in private) work than the private organizations, they have outsourced less of the operations (31% in public, 41% in private) and user support (21% in public, 29% in private). Other important aspects to take into account are the rise of new delivery models (e.g. OSS) not only for infrastructure applications, and the trend towards using cloud infrastructures for the operations of ICT-solutions. Systems span to a larger degree organizational boundaries, and include an increasing number of sources for input of data including sensors proving large amount of data.

A number of partly related needs can be identified in this landscape that must be addressed in future business information systems:

- The support of end-to-end engineering process (including full life-cycle support)
- Integration across organizations and nations
- Systems being provided by ecosystems of providers
- Event-oriented systems utilizing the internet of things

We will in the next section go into detail on more detail on each of these areas, discussing in particular the application of modeling techniques.

2 Towards Future Business Information Systems

2.1 The Support of End-to-End Engineering Process (Including Full Life-Cycle Support)

Even with an apparent shift to a service economy, manufacturing and engineering remains vitally important for the economy. According to [31], “before the present economic crisis, manufacturing contributed some 17.1% of GDP and accounted for some 22 million jobs (2007) in Europe”. Taking sectors into account which are directly related to manufacturing (e.g., transport), [31] even names a share of 47% of GDP. However, the influence of the economic crisis on industry has decreased output by around 20%, while global competition is growing dramatically. This leads to increasing pressure. Moreover, we witness new trends and paradigms such as sustainable manufacturing and mass customization. Consequently, the manufacturing industry is facing significant structural changes.

The key enabler for coping with these changes will be ICT, due to its strong impact on innovation and productivity (cf. [8], [9] and [35]). The current ICT for manufacturing landscape is characterized by scattered data formats, tools and processes dedicated to different phases in the product lifecycle: in the concept phase of a product, often simple tools like MS PowerPoint are used, while later on it may be specialized CAx solutions, PLM and ERP systems, etc. Moreover, the flow of information is closely aligned to the product lifecycle (i.e., information from the design phase goes into the manufacturing phase, but the opposite direction as well as the flow of user feedback into design are often broken or neglected). Due to the diversity of tools and data formats, manufacturing struggles to cope with new trends in this area. For example, both the trend to mass customization and the demand for increased sustainability require a tight integration of the design, manufacturing and usage phases of a product, which is currently not in place. The rise of web 2.0 leads to precious information, manifested in web 2.0 channels like blogs and Facebook groups, created directly by prospective or existing users of a given product or service. But this sort of information having limited impact to the design or manufacturing phase of a product.

To summarize: What is clearly missing in the current ICT landscape for manufacturing and engineering is an integrated, holistic view on data, persons and processes across the full product lifecycle. As experiences of the past show, a tight integration of all tools used throughout a product lifetime is not feasible. A model-based approach to address this is the application of AKM (Active Knowledge Modeling) [30].

AKM requires a new way of representing knowledge as visual models, where complex, rigid, software-oriented languages are replaced by simple and agile domain concepts. A model is a representation of some aspects of the real world entities and phenomena, as interpreted by some actor(s). A model is active if it also influences the reality that it represents. Knowledge is held by people, so knowledge modeling languages should be based on human communication, sense-making and learning, rather than computer and software concepts.

Activation, the process whereby a model influences reality, cannot be solely based on automated execution. Instead, users must be supported by flexibly interpreting the models and acting upon them, in the situations that arise. This principle is called interactive activation. It implies partial automation, where the automation boundary may be moved by the users (see also 2.4.4 below). The more precise and detailed a model is the more automatic execution it supports. However, exception handling requires that the users be allowed to "open up" the model and change the default, automatic interpretation of it.

Knowledge architectures consist of knowledge explicitly represented in structured models, and of the mental views of the people involved in creating and using these models. Knowledge is explicitly represented as information and data structures. Data consists of symbols used for conveying information and knowledge. Data becomes information when its meaning is interpreted by some actor. We thus see data as a one-dimensional representation, a stream of symbols. Information adds a second dimension that reflects the meaning of the original data, often called meta-data. Knowledge implies a justification of the information or that the information guides action. Knowledge establishes structures and relationships between information elements, and uses them to manage dependencies. Knowledge representations must thus possess at least three dimensions: data, its meaning and the structure, justifications, and actions that the knowledge results in. In order to support reflection on knowledge, a fourth dimension is needed. Reflection on knowledge is required for e.g. learning, knowledge management, design, innovation, collaboration, creation of shared understanding, and other creative tasks. We refer to representations that contain four reflective dimensions as knowledge spaces. These concepts are illustrated in Figure 1.

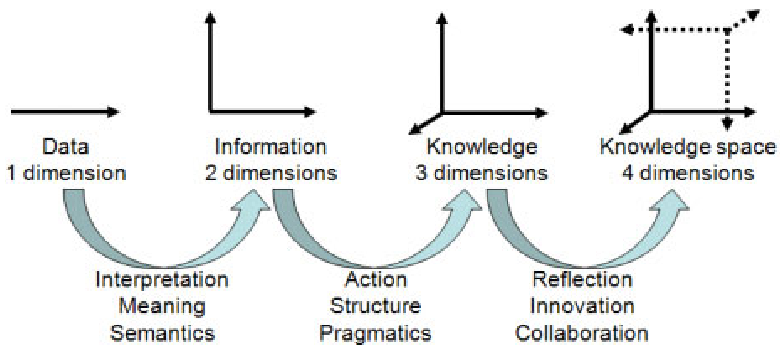


Fig. 1. Dimensions in Modeling

- AKM supports human processes such as communication and learning.

In software systems, the second dimension is typically represented as program code that defines how the data is manipulated, stored and presented to the user. Among humans, the second dimension can be illustrated by the capability to understand a certain language, such as English. If you do not understand the language, speech becomes incomprehensible data.

Few computerized systems are really knowledge-based. Their data structures and program code are fixed. By using the system, humans can bring in the extra dimensions of interpretation and reflection needed for knowledge and learning. However, these additional dimensions cannot be reflected back into the computerized system as updated data structures or program logic. There is no easy way to affect the behavior of the system other than manually changing the code. Thus, reflection, knowledge and learning cannot be shared among the people using the system. What a user can learn from the system is limited to the two dimensions that were coded into the system from the start. The knowledge space of software consists of two explicit dimensions (data, code) and two mental views possessed by different roles (programmers, users).

- AKM extracts knowledge from enterprise systems.
- AKM provides the possibility to affect the behavior of systems by users.

The concept of Model-Generated Workplaces (MGWP) Figure 2, can be used to make the models available in a tailorable way. MGWP is a working environment for the business users (including engineers) involved in running the business operations of the enterprise. It is a user platform that provides the graphical front-end for human users to interact with software services supporting their day-to-day business activities. The workplace can be tailored to meet the specific requirements of different roles or persons within an enterprise, providing customized presentation and operation views based on data in the enterprise systems.

- AKM supports customized role-based views of enterprise data.

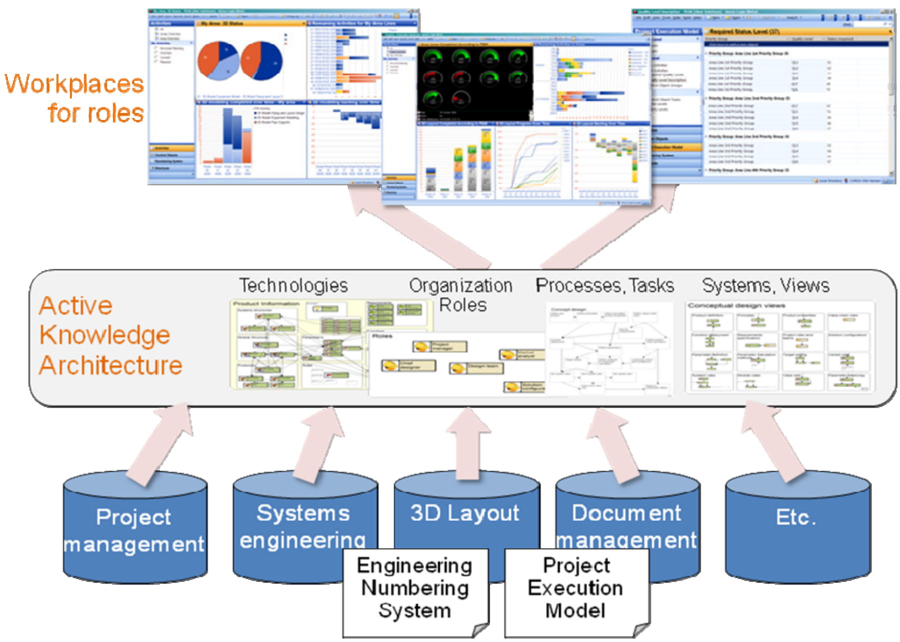


Fig. 2. Operational view of a model-generated workplace

An overall solution will benefit from a combination of this type of approach with semantic web-oriented approaches when the possibility of formalizing the knowledge representations is beneficial.

2.2 From Development in One Organization to Integration Across Organizations

As emerging technologies permit new ICT system architectures, new and promising opportunities in the context of Business Process Management (BPM) arise. Based on globalization trends, new challenges pop up, particularly when multi-national companies need to coordinate their local business units in order to serve other multi-national companies in an integrated fashion. In an earlier reported case of a certification company [20, 21], there was a need to standardize the processes of the company's national branches in order to build a common image of the organization (both inward and outwards), and to support the certification of the cross-national processes of their multi-national customers, but at the same time adhere to national and cultural rules and expectations.

In such a context the integration of common technologies like mobile devices, techniques from the ubiquitous computing context, so called smart environments as well as the increasing use of sensor network technologies for the collection of process-relevant data and the application of service-oriented architectures (SOA) as well as Web 2.0 technologies can improve the flexibility of inter-corporate BPM [38] and thus increase the effectiveness and efficiency of business processes in inter-corporate value chain networks. Furthermore the options of action for the human actors involved increase. Figure 3 illustrates a collaborative scenario in a value chain network applying the technologies mentioned. Scenarios like these are not only of importance in business, but also in the public administration area as described in the last EU Ministerial Declaration on eGovernment [33] which emphasizes the special need to develop and improve cross-border eGovernment services, making it easier for businesses and citizens to operate in and across any EU-member state.

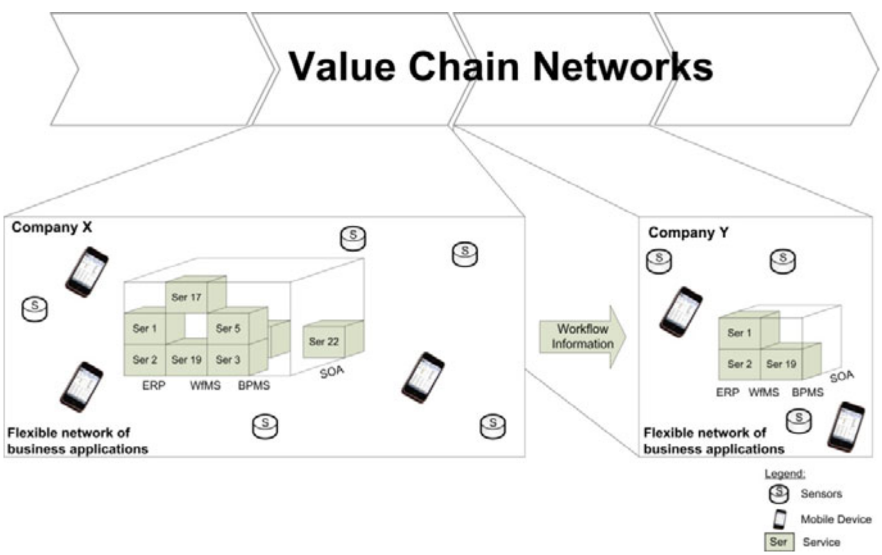


Fig. 3. Collaborative scenario in Value Chain Networks

The above scenario shows four important *trends*:

1. Processes are *increasingly interconnected* and it makes often no sense to look at a single process in isolation,
2. The *number of processes* an organization has to cope with *is rapidly increasing* (large organizations have hundreds of processes which need to be managed),
3. Modern technology is generating unprecedented streams of event data representing the states of different processes (sensor data, RFID data, remote logging, remote services etc.) (described further in 2.4) and
4. Different devices are used to access the BPM System (BPMS) in different situations necessitating a flexible multi-channel support influencing which parts of the workflow are available in which manner depending on the context of use.

Based on these trends and the application of the mentioned technologies on the one hand the enterprises' agility and the handling of more and more dynamic business conditions can be improved. On the other hand Business Process Management becomes more and more complex. The *reasons for this complexity* are manifold [15]:

1. The range of inter-corporate collaborative business processes,
2. The number of organizational units involved in a business process,
3. The need to manage and control mobile actors in business processes,
4. The need to control person-machine and machine-machine interactions,
5. The interdependencies in sensor networks, and
6. The need to manage services in a business process applying SOA etc.

The handling of this complexity which is generated by many thousands of process types, process instances and process events is a new challenge in BPM, and thus also modeling of business processes, where one need to address BPM-in-the-large [15]. In comparison with "BPM-in-the-Small" these approaches support a better complexity handling in real-life BPM scenarios, both tapping automation potentials arising from the vast amount of event data generated by the Internet of Things as well as assisting human actors in designing and coordinating complex real-life business processes. Table 3 summarizes important characteristics and contours of BPM-in-the-Small and BPM-in-the-Large.

2.3 From Systems Being Provided by Individual Organizations to Systems Being Provided by Ecosystems

As indicated above, the trend has been towards systems being developed and evolved further away from the users of the system. We see a development in the direction of systems to a larger degree being supported by *virtual communities* of nomadic, human/organizational *actors*, co-working on partially shared digital artifacts [2]. The term *'digital ecosystem'* has recently been used to generalize such communities, emphasizing that their actors constantly interact and cooperate with other actors in both local and remote ecosystems. Such systems are characterized by self-organization, scalability and sustainability, providing both economic and social value. Examples are communities for Open Source (OSS) and Creative Commons, social media networks as in Facebook, blogs and computer games, virtual organizations with scattered employees, or voluntary groups of citizens or academics.

Table 3. Characteristics of BPM-in-the-Small and BPM-in-the-Large, inspired by [33]

Characteristic	BPM-in-the-Small	BPM-in-the-Large
1. Temporal and spatial extent of process definitions	focus on one or few cooperating organizations	focus on comprehensive value networks containing lots of organizations
2. Structuring of processes	often a priori fixed process structure	more flexible process structures which have to be adapted to changing situations
3. Number of different process types	often a priori fixed number of defined core and support processes	increasing number of processes through configurable process models and large process portfolios
4. Dependency between processes	often few dependencies	Many dependencies
5. Number of process stakeholders	low number of stakeholders in single processes	high number of stakeholders in inter-organizational processes
6. Dynamics of process change	often stable and seldom changed process structures	often adapted and flexibly configured process structures
7. Heterogeneity of applied modeling notations	low heterogeneity	high heterogeneity
8. Number of different coordinating institutions of cooperative processes	low	high
9. Dominant planning direction	often central organization, top-down	decentralized organization, bottom-up
10. Architecture for BPMS	often monolithic	loosely coupled BPM services

However, the existing digital ecosystems have limited scope, various degree of transparency, insufficient capabilities for search and evaluation of useful quality artifacts from the huge and ever-evolving Internet, and none does fully support a wide range of shared artifacts from a wide range of actors. There are two main variants of digital ecosystems; *content ecosystem and software ecosystems.*

Content ecosystems are networks that deals with creation and sharing of artistic and intellectual artifacts**.** The impact of ICT on participative and democratic processes and on creativity is already here, and will continue to grow with the increasing diffusion of web-based social networking and user generated content and services. Internet already allows highly visual and multimodal interactions, and these interactions will become richer. Earlier work under the title digital ecosystems, such as OPAALS (http://www.opaals.org/) addresses only parts of this area.

S*oftware ecosystems* are "*a set of businesses functioning as a unit and interacting with a shared market for software and services, together with relationships among them. These relationships are frequently underpinned by a common technological platform and operate through the exchange of information, resources, and artifacts*" [32]. See also work on software ecosystems for product familie*s* [3], more general software systems [16], and guidelines for using such ecosystems [17]. For instance within OSS [10] hundred thousands of co-worked software "components" are freely available, often via portals like http://sourceForge.net. The quality is variable and often poorly documented. Yet, 50% of Norwegian ICT companies now integrate OSS components into their own applications, and 10% also contribute back to the net [13]. Traditional customers - like municipalities - cooperate to provide improved e-services for their inhabitants. And end-users - even kids - are becoming their own developers.

To address combined digital content and software ecosystems, there must be substantial and concerted improvements of the state-of-the-art in three traditionally unrelated and partially isolated research areas:

- Enterprise architecture and enterprise modeling
- New business models
- Data management

The kind of modeling we are looking on in our work, in particular applies to the first two areas. Public and private organizations are becoming less self-sufficient and increasingly dependent on business partners and other actors, e.g. by outsourcing non-core activities. However, when such cooperation moves beyond simple buying and selling of goods and well-defined services, there is a need for a flexible infrastructure that supports not only information exchange, but also knowledge creation, evolution and sharing across a myriad of cooperation networks that tend to work in a bottom-up and rather improvised way. Within many organizations, it has become customary to develop enterprise architectures [28]. An ecosystem architecture takes the ideas of enterprise architecture to a higher level of abstraction, looking upon the support of a more fluid landscape of business actors providing and consuming services for information systems support in an organizational setting. In this way it extends the process perspective in BPM-in-the-Large to a wider setting. This demands a new approach to enterprise integration. User-initiated software applications and enterprise mash-ups should be based on active knowledge modeling and supports learning [30].

For this to function, one must support more open business models. In this, one must consider financial success, sustainability, competition, copyright and licensing, and the impact on work processes, leadership, internal coordination, work processes, strategy and planning. The *Open innovation* approach [5,6] is often chosen as a basic cooperation mechanism. Companies should allow more free ("open") import and export of ideas and knowledge concerning products, processes and business models that flow between organization and their environments. Indeed, more openness (less proprietary copyrights and patents) will provide a larger set of possible business opportunities, which will lead to decisive competitive advantages. However, the problems connected to IP and revenue sharing must be considered. Furthermore, an open innovation strategy must be reflected not only in the business models, but also by revised behavior (process practice) and in new thinking patterns. The cooperating organizations contribute to a complex value network, whose success criteria go beyond traditional value chain analysis, in order to capture the complex inter-dependencies between companies and between technological architectures and business models. The SEM language [17] attempts to analyze the business along customer-supplier lines. Furthermore, the E^3value model [12] describes value-generation among partners in a value network.

2.4 From Transaction-Oriented Systems to Event-Oriented Systems Utilizing the Internet of Things

Under the last decade large changes have taken place on the ICT-arena. What previously was termed convergence (between the telecommunications world, IT, media, and later also power systems) is now emerging in practice, propelled by several simultaneous trends that was partly discussed in the introduction:

1. The continuing miniaturization of computing resources making it possible to perform computing at some level everywhere, by any device.
2. The availability of high-bandwidth access to computing resources in an increasing number of places.
3. The infrastructure being built up for utilizing remote computing resources (these days often presented under the term 'cloud computing').
4. More power-efficient solutions. Many battery-operated devices can last a year or more, and passive solutions used by RFID and NFC (Near Field Communication) are powered by readers. Parasitic energy harvesting devices that extract small amounts of energy from the environment can power sensors where normal power solutions are not available.

The emergence of the IoT will lead to a world in which countless everyday objects are interconnected and have their own computational power. These “living” objects will be able to monitor and change their environment via sensors and actuators, and to interact and collaborate with each other and with the people around them. Although more accessible, to quickly produce innovative applications and services for this setting is not an easy task. Information systems need to change from transaction-orientation to event-orientation, using event-driven architectures[1] (EDA) reflecting the nature of IoT applications. These application are characterized by being:

- Collaborative, that is including numerous active components that behave concurrently, running on distributed devices, and collaborate to provide a desired functionality.
- Event-driven and reactive, that is continuously reacting on a large number of events from the physical environment, from users and from other parts of the system.
- Dynamic and adaptive, meaning that applications, associations and collaborations are dynamically established and configured depending on the current situation at run-time. The information gathered is often used to support human decision-making rather than to only automate a predefined process. Interaction with human users is essential both to use the systems efficiently, and to effectively improve the systems over time.

The survey of current and future research in this area is structured according to the following areas:

1. Rapid application development
2. Rapid application deployment

[1] We use the term Event Driven Architecture in a wider meaning than event pipe-and-filter architectures here. We include the general class of reactive systems consisting of loosely coupled, distributed, components that interact by passing messages representing events among each other.

3. End-user tailoring of application and services
4. Collaborative IoT applications

2.4.1 Rapid Application Development

Current research and development at NTNU [18] have built a solid foundation to enable rapid development, building upon the following key principles:

- Reuse and compositionality: We have developed a new kind of model-based building blocks that are collaborative and event-driven. Their structure fosters reuse and highly effective possibilities for adaptation and composition.
- Requirements and domain languages: Adequate languages specific to a certain domain can capture specific problems of a domain better and faster, and can be the basis for the development of executable prototypes.
- Automatic analysis and applied formal methods: Formal methods for analyzing specifications thoroughly have existed for a long time, but they are not used effectively in practice. More automated approaches executing in the background can change this.
- Automatic implementation and model-driven development: Automating these processes saves manual and time intensive labor, but also reduces the amount of errors that are introduced in these phases dramatically. This requires high-quality models to begin with.

With Arctis, NTNU has built an entirely new type of tool for the incremental and compositional development of event-driven systems. It is based on the concept of special building blocks from domain-specific libraries, expressed by UML and Java. On average, we have measured reuse proportions of 71 % [19], which leads to a considerable decrease in development time. For research, Arctis is especially interesting due to its analytic capabilities that are based on model checking, and enable an automatic check of systems [18]. With the integrated code generators, Arctis can develop systems for the commercially available IoT-platforms, mobile phones with Android, and embedded systems.

2.4.2 Rapid Application Deployment

A number of IoT systems and applications already exist in real world settings (e.g. for surveillance purposes), but they are all based on proprietary, non-standard solutions, and are deployed with a specific domain in mind. Application development for such systems, as well as interoperation with other systems is difficult due to the closed nature of such solutions and the variety of technologies that are used. Recent research has therefore focused on IoT integration frameworks, which provide common interfaces for heterogeneous sensors and actuators, and ease their deployment and management. Examples are GSN, SWE and SENSEI: Global Sensor Networks (GSN) [1] is an open source middleware project. It enables fast and flexible deployment and interconnection of sensor networks. The Sensor Web Enablement (SWE) [4] is an initiative from the Open Geospatial Consortium that aims at creating open standards for plug-and-play of sensors and actuators, such that they are accessible and controllable via the Web. SENSEI [36] is a recent European research project. Its goal is to create a common, global framework that integrates different sensors and actuators and makes them available to services and applications via universal service interfaces.

IoT integration frameworks, by providing a common interface for different sensors and actuators and a communication "bus" for event driven applications, facilitate the development of applications on top of them. However, application development is still challenging due to the intrinsic characteristics of this type of applications, such as context-sensitivity and dynamic structure. While there have been some proposals for model-driven development approaches of such applications (e.g. [27], [37]), the main challenges remain still unaddressed.

2.4.3 End-User Tailoring of Application and Services

Intelligent objects and devices are becoming part of the environment where people live. The more pervasive the IoT becomes, the greater need users have to tailor the computing activities that take place around them. For some people the large number of devices and services represent a problem: how to manage the complexity? For others this provides opportunities for tailoring things to exactly what one wants. Service discovery, tailoring and composition should not require expert knowledge, but be possible for ordinary users with sufficient interest and skill. One project looking at this problem area is Ubicompforall [11].

2.4.4 Collaborative IoT Applications

A large number of application areas for the more generic technologies described above can be identified. We here in particular look on mobile, collaborative applications and services for application in the health sector. Any healthcare system aims to monitor the health of the population and to mitigate its threats by collecting and analyze data about health problems, and subsequently reacting on them. As such, a healthcare system can be described as a distributed, event-driven, reactive problem solving machinery. Having, as its base, a set of interconnected, semi-autonomous care providing units, one of the most crucial functions of a healthcare system is the mechanism for distributing health problems to the most appropriate problem solver. From assisted self-help groups to nursing homes, from the local doctor to the more distant, technologically advanced university hospital, cost effective care requires an efficient mechanism for shuffling medical problems to the most appropriate problems solvers.

What is the most appropriate level of problem solving might be a matter of healthcare policy, but mostly is dictated by technological advances and decisions to bring these to work. As new technologies tend to come with a heavy price tag, some medical technology investments must be accompanied by a redirection of the flow of patients to the care unit that decided to implement the technology. Other medical technologies establish a window of opportunities for new patient groups.

Hence, the establishing of new healthcare technologies bring about profound changes in the flow of problems and patients within the system, changes that has brought today's information-exchange and communication system to the limit and must be addressed by event-driven rather than transaction-driven systems.

Clinicians face data processing challenges in decision-making situations. Of these, there are two types:

1. *Deciding on medical acts — what to do with the patient.*
2. *Deciding on coordination acts — which patient to work on next.* Knowing what has been going on in the clinical process enables clinicians to adapt their plans and coordinate their work with that of others. In addition to patient data, these

decisions are informed by data about what other personnel are doing and which resources (rooms and equipment) are in use.

In the ongoing COSTT project (www.costt.no) one focus on the question how the second challenge can be supported by ICT. The digital information infrastructure that now is emerging in many hospitals will become a vast resource. But to utilize this "ocean" of data, there will be an increased need for data filtering, processing and visualizing techniques and for systems that assist upon the decisions of clinicians and coordinators by providing useful data visualizations. Another relevant area is the efficient monitoring of persons outside the hospital in potential need of care, but in no interest in surveillance.

Traditional information systems often try to automate as much as possible based on static knowledge of the situation to be supported. Already in the 80ies, it was realized that many processes where better to use IT for informing workers rather than for automating their work, and one have since then described a number of archetypical process types that can be reasonably supported by IT. The different process types decide the extent to which the underlying technology can be based on hard-coded, predefined, evolving or implicit process models. This gives a number of development approaches as illustrated in Fig. 4. On one extreme; systems are manually coded on top of a traditional runtime environment, and on the other loose, enterprise models are used directly to generate and evolve solutions. In between these, we have the approaches typically described in model-driven development and BPM. The use of BPM and Workflow technology is typically based on an existing execution platform for execution of the process solution.

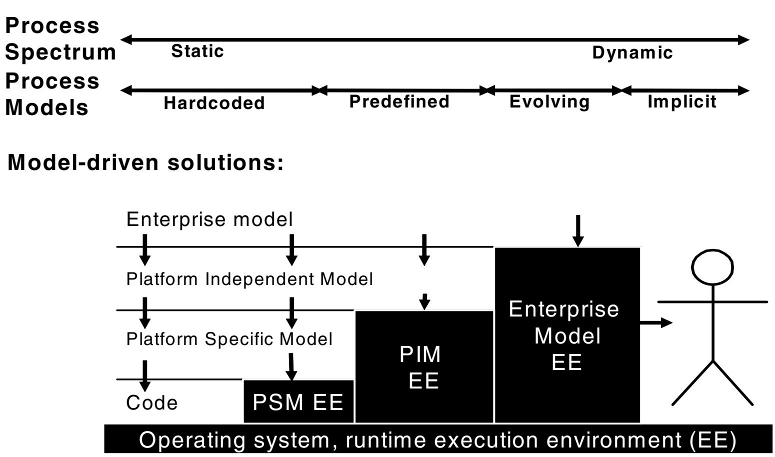


Fig. 4. Overview of different execution environment for different process models

Workflow traditionally focus on static process support where one generally separate process definition (articulation, modeling) from process enactment (practice, activation). These activities are supported by different tool components, and performed by different roles. As the process is enacted, the roles are filled with individual people (process participants), and they seldom have knowledge outside their own responsibilities.

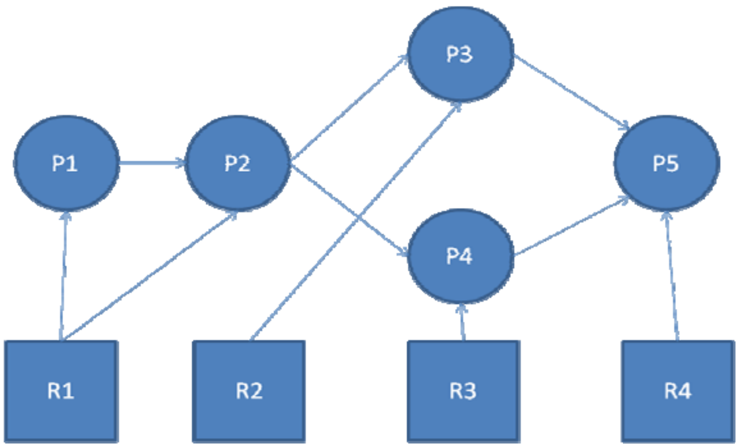


Fig. 5. Role-restricted scope of information access

This is illustrated in Fig. 5 above where those in role R1 have overview of task necessary for P1, and P2, R2 for P3 etc. A more resilient solution would be to let those in a role (e.g. Role R3 which do P4) also have access to information relevant for P3 (parallel task), P2 (prior task) and P5 (dependant task).

The limited success of WMS in supporting knowledge intensive and cooperative work, has partly been attributed to lack of flexibility. Most work within this area looks at how conventional systems can be extended and enhanced, how static workflow systems can be made *adaptive*. Research challenges for adaptive workflow include:

- Controlled handling of exceptions
- The dynamic change problem.

Some also investigate more far-reaching adaptability including late modeling during enactment and local adaptation of particular workflow instances. Most researchers in this area recognize that change is a way of life in organizations, but a basic premise is still that work is repetitive and can be relatively completely prescribed. Within the community, an understanding seems to have emerged that change requires process definition and process enactment to be intertwined. But still, most research on adaptive workflow is based on the premise that the enactment engine is solely responsible for interpreting the workflow model. In other words, users contribute by making alterations to the model, not by interpreting any part of the model. Thus, the model must be formally complete to prevent ambiguity and deadlock from paralyzing the process. Process definition is the work of process experts, while process participants perform the work.

Interactive workflow which we in this paper have discussed as part of the AKM philosophy steps beyond this limitation

Similar scenarios where a lot of potential information from sensors of different kinds can be gathered to be used for supporting (not automating) decision making among humans can be found in the energy sector, as part of the area called Smartgrid. A smart grid is a modernized electricity network that delivers electricity from suppliers to consumers using two-way digital technology to control appliances at consumers' homes. The overall goal of smart grids is to save energy, reduce cost and increase reliability and transparency. Smart grids are being promoted by many governments as a way of addressing energy independence, global warming and emergency resilience issues. Smart grids make the traditional electricity grid 'intelligent' by deploying sensors at various points in the electric supply chain. The sensors provide a feedback mechanism to both customers and providers of energy.

For customers, the Smart Grid supplies granular information about electricity consumption and helps them make cost effective choices about their energy use. Providers can also leverage this information to create and distribute energy in a more efficient manner.

3 Concluding Remarks

From the above descriptions, we see that the technical challenges and opportunities with the future internet gives new challenges and opportunities for business information systems on many levels, but that all of these give new challenges and opportunities for model-based techniques such as BPM, MDA (as exemplified in Arctis), enterprise modeling, value modeling and AKM. In a way many of the core problems are not new, e.g. how to deal with event-based systems have been investigated in the conceptual modeling area for quite some time. Even if the use of modeling need to be extended and improved, general categories underlying discussions on quality of models as described in [24,25] remains relevant, although need to be adapted to e.g. quality of interactive models vs. more static models [23,30].

Acknowledgements. The material in this paper has been developed in collaboration with a large number of persons, e.g. relative to the development of a number of research proposals over the last years. It is often hard in such processes to pinpoint each person's contributions, but the following should at least be credited for being involved in the scientific discourse related to the materiel presented in this article (in alphabetic order): Wil van der Aalst, Rolf Bræk, Jacopo Cassina, Reidar Conradi, Arild Faxvaag, Peter Fettke, Jon Atle Gulla, Jacqueline Floch, Hans Christian Haugli, Peter Herrmann, Arild Herstad, Constantine Houy, Knut Eilif Husa, Letitzia Jaccheri, Thomas Jelle, Håvard Dingstad Jørgensen, Kirsti Kierulf, Frank Kraemer, Kjetil Kristiansen, Anja Kugel, Frank Lillehagen, Peter Loos, Geir Melby, Kjetil Nørvåg, Tonje Osmundsen, Sobah Abbas Pettersen, Per Morten Schiefloe, Guttorm Sindre, Ole Ivar Sivertsen, Atle Sægrov, Pieter Toussaint, Alf Inge Wang.

References

1. Aberer, K., Hauswirth, M., Salehi, A.: Infrastructure for data processing in large-scale interconnected sensor networks. In: Proc. 8th Intl. Conf. on Mobile Data Management (MDM 2007). IEEE CS, Germany (2007)
2. Ali Babar, M., Dingsøyr, T., Lago, P., van Vliet, H. (eds.): Software Architecture Knowledge Management - Theory and Practice Establishing and Managing Knowledge Sharing Networks. Springer, Heidelberg (2009)

3. Bosch, J.: Architecture Challenges for Software Ecosystems. In: Proc. Fourth European Conference on Software Architecture (ECSA) - Companion Volume, Copenhagen, Denmark, pp. 93–95. ACM Press, New York (2010)
4. Botts, M., Percivall, G., Reed, C., Davidson, J.: OGC Sensor Web Enablement: Overview And High Level Architecture. Open Geospatial Consortium White Paper (December 2007)
5. Chesbrough, H.: Open innovation. The new imperative for creating and profiting from technology. Harvard Business School Press, Boston (2003)
6. Chesbrough, H.: Open Services Innovation. Rethinking your business to growth and compete in a new era. Jossey-Bass, San Francisco (2011)
7. Davidsen, M.K., Krogstie, J.: A longitudinal study of development and maintenance. Information and Software Technology 52, 707–719 (2010b)
8. Digital Agenda (2010), `http://ec.europa.eu/information_social/digital-agenda/index_en.htm`
9. EU (2020), `http://ec.europa.eu/eu2020/index_en.htm`
10. Fitzgerald, B.: The Transformation of Open Source Software. MIS Quarterly 30(3), 587–598 (2001)
11. Floch, J., Herrmann, P., Khan, M.U., Sanders, R., Stav, E., Sætre, R.: End-User Service Composition in Mobile Pervasive Environments. In: 2nd International Workshop on EUD4 Services in Brindisi (June 2011)
12. Gordijn, J., Yu, E., van der Raadt, B.: E-service design using i* and e3value modeling. IEEE Software 23(3) (2006)
13. Hauge, Ø., Ayala, C., Conradi, R.: Adoption of Open Source Software in Software-Intensive Industry - A Systematic Literature Review. Information and Software Technology 52(11), 1133–1154 (2010)
14. Holgeid, K.K., Krogstie, J., Sjøberg, D.I.K.: A study of development and maintenance in Norway: Assessing the efficiency of information systems support using functional maintenance. Information and Software Technology 42, 687–700 (2000)
15. Houy, C., Fettke, P., Loos, P., van der Aalst, W.M.P., Krogstie, J.: BPM-in-the-Large – Towards a higher level of abstraction in Business Process Management. Paper Presented at the Accepted at GISP under WCC (2010)
16. Hunink, I., van Erk, R., Jansen, S., Brinkkemper, S.: Industry taxonomy engineering: the case of the European software ecosystem. In: Proc. Fourth European Conference on Software Architecture (ECSA) - Companion Volume, Copenhagen, Denmark, pp. 111–118. ACM Press, New York (2010)
17. Jansen, S., Finkelstein, A., Brinkkemper, S.: A sense of community: A research agenda for software ecosystems. In: Proc. 31st International Conference on Software Engineering (ICSE), New and Emerging Research Track - Companion Volume, Vancouver, Canada, May 16-24, pp. 187–190 (2009)
18. Kraemer, F.A., Slåtten, V., Herrmann, P.: Tool Support for the Rapid Composition, Analysis and Implementation of Reactive Services. The Journal of Systems and Software 82(12) (December 2009)
19. Kraemer, F.A., Herrmann, P.: Automated encapsulation of UML activities for incremental development and verification. In: Schürr, A., Selic, B. (eds.) MODELS 2009. LNCS, vol. 5795, pp. 571–585. Springer, Heidelberg (2009)
20. Krogstie, J., Dalberg, V., Jensen, S.M.: Harmonising Business Processes of Collaborative Networked Organisations Using Process Modelling. In: Virtual Enterprises and Collaborative Networks. IFIP, vol. 149, pp. 81–88. Springer, Boston (2004)

21. Krogstie, J., Dalberg, V., Jensen, S.M.: Process modeling value framework. In: Manolopoulos, Y., Filipe, J., Constantopoulos, P., Cordeiro, J. (eds.) ICEIS 2006. LNBIP, vol. 3, pp. 309–321. Springer, Berlin (2008)
22. Krogstie, J., Jahr, A., Sjøberg, D.I.K.: A longitudinal study of development and maintenance in Norway: Report from the 2003 investigation Information and Software Technology 48, 993–1005 (2006)
23. Krogstie, J., Jørgensen, H.D.: Quality of Interactive Models. Paper Presented at the First International Workshop on Conceptual Modelling Quality, IWCMQ 2002 (2002)
24. Krogstie, J., Sindre, G., Jørgensen, H.D.: Process Models as Knowledge for Action: A Revised Quality Framework. European Journal of Information Systems 15(1), 91–102 (2006)
25. Krogstie, J., Sølvberg, A.: Information systems engineering - Conceptual modeling in a quality perspective. Kompendiumforlaget, Trondheim (2003)
26. Krogstie, J., Sølvberg, A.: Software maintenance in Norway: A survey investigation. In: Muller, H., Georges, M. (eds.) Proceedings of the International Conference on Software Maintenance (ICSM 1994), pp. 304–313. IEEE Computer Society Press, Los Alamitos (1994)
27. Kushwaha, M., Amundson, I., Koutsoukos, X., Neema, S., Sztipanovits, J.: OASiS: A Programming Framework for Service-Oriented Sensor Networks. In: Procs. of 2nd Intl. Conf. on Communication Systems Software and Middleware (COMSWARE 2007). IEEE CS, Los Alamitos (2007)
28. Lankhorst, M.: Enterprise Architecture at Work. Springer, Heidelberg (2009), ISBN 978-3-642-01309-6
29. Lientz, B.P., Swanson, E.B., Tompkins, G.E.: Characteristics of application software maintenance. Communications of the ACM 21(6), 466–471 (1978)
30. Lillehagen, F., Krogstie, J.: Active Knowledge Modeling of Enterprises. Springer, Heidelberg (2008) ISBN 978-3-540-79415-8
31. `http://ec.europa.eu/enterprise/policies/industrial-competitiveness/economic-crisis/files/eu_manufacturing_challenges_and_opportunities_en.pdf` (2010)
32. Messerschmitt, D.G., Szyperski, C.: Software Ecosystems: Understanding an Indispensable Technology and Industry. MIT Press, Cambridge (2003), ISBN 0-262-13432-2
33. Ministerial Declaration on eGovernment, Malmø, Sweden 5th Ministerial eGovernment conference (2009)
34. Nosek, J.T., Palvia, P.: Software maintenance management: Changes in the last decade. Journal of Software Maintenance 2, 157–174 (1990)
35. OECD, Policy Responses to the Economic Crisis: Investing in Innovation for Long-Term Growth. OECD report (June 2009), `http://www.oecd.org/dataoecd/59/45/42983414.pdf`
36. SENSEI project, `http://www.ict-sensei.org`
37. Ulbrich, A., Weis, T., Mühl, K.G.: Application Development for Actuator- and Sensor-Networks. In: Proc. of GI Workshop on Sensor Networks, ETH Zurich, Switzerland (March 2005)
38. Vanderhaeghen, D., Fettke, P., Loos, P.: Organizational and Technological Options for Business Process Management from the Perspective of Web 2.0 - Results of a Design Oriented Research Approach with Particular Consideration of Self-Organization and Collective Intelligence. Business & Information Systems Engineering 2, 15–28 (2010)

Next Generation of Modelling Platforms

Dimitris Karagiannis and Niksa Visic

University of Vienna, Knowledge Engineering Research Group, Brünnerstr. 72,
A-1210 Vienna, Austria
{dk,nv}@dke.univie.ac.at

Abstract. Future enterprise systems require an elaborate conceptual foundation that promotes a tight mutual alignment between information systems and business to effectively support business operations and managerial decision-making. Thus a growing number of groups around the world show interest in modelling methods – either standard or individual ones – that satisfy the requirements of their domain and comply with the conceptual foundations. In order to analyze modelling methods in different domains, we introduce a generic modelling method specification framework that describes modelling methods on three major parts: (i) the modelling language that describes the syntax, semantics and notation, (ii) the modelling procedures that describe the methodology as well as (iii) algorithms and mechanisms that provide "functionality to use and evaluate" models described by a modelling language. Simultaneous to the raise of modelling methods there is a need for re-use, integration or combination of different modelling methods. The metamodelling approach is considered to provide the required concepts and mechanisms to combine different modelling methods for the so called Hybrid Method Engineering. The Next Generation Modelling Framework (NGMF) supports Hybrid Method Engineering, both on a conceptual and on a technical integration level. On conceptual level the NGMF provides mechanisms to encapsulate modelling methods and enable the hybrid use of different modelling languages. On technical level the Next Generation Modelling Platform provides functionality for modelling method engineering and modelling method application. The organizational framework is provided by the Open Model Initiative (OMI), which supports users in realizing and applying this approach based on three pillars: community, projects and foundations.

Keywords: Metamodelling Platforms, Hybrid Modelling, Modelling Methods, Modelling Languages, ADOxx.

1 Introduction

Modern day system developers have some serious problems to contend with. The systems they develop are becoming increasingly complex as customers demand richer functionality delivered in ever shorter timescales. To add to that, nothing stays still: today's "*must have*" technology rapidly becomes tomorrow's legacy problem that must be managed along with everything else. Languages are the primary way in which system developers communicate, design and implement systems. Languages provide abstractions that can encapsulate complexity, embrace the diversity of technologies and

J. Grabis and M. Kirikova (Eds.): BIR 2011, LNBIP 90, pp. 19–28, 2011.

design abstractions, and unite modern and legacy systems. The benefit of *metamodelling* is its ability to describe these languages in a unified way. This means that the languages can be uniformly managed and manipulated thus tackling the problem of language diversity. For instance, mappings can be constructed between any number of languages provided that they are described in the same metamodelling language. Using metamodels, many different abstractions can be defined and combined to create new languages that are specifically tailored for a particular application domain [19]. As a result productivity is greatly improved.

Uses for a metamodel can be summarized as follows: define the syntax, notation (sometimes also called visual or graphical syntax) and semantics of a language, explain the language, compare languages, specify requirements for a tool for the language, specify a language to be used in a meta-tool, enable interchange between tools, enable mapping between models.

1.1 Basic Definitions

The notion of model goes beyond the narrow view of semi-formal diagram thus requiring much more precise definitions. The following definitions help us in understanding the concept of SUS, model, conceptual model, metamodel, and meta-metamodel.

A *System under Study* (SUS) is a delimited part of the world considered as a set of elements and interactions. A *model*, representation of a given SUS, is a directed multigraph that consists of set of nodes, a set of edges, and a mapping function between nodes and edges, where nodes may be connected with more than one edge, and is such that its reference model is a metamodel [24]. *Conceptual model*, also known as domain model, represents concepts (entities) and relations between them, and is independent of design or implementation concerns [26]. The aim of a conceptual model is to express the meaning of terms and concepts used by domain experts to discuss the problem, and to find the correct relationships between different concepts. The conceptual model attempts to clarify the meaning of various, usually ambiguous terms, and ensure that problems with different interpretations of the terms and concepts cannot occur. A *metamodel* is a model such that its reference model is a meta-metamodel [24]. In its broadest sense, a metamodel is a model of a modelling language, and it must capture the essential features and properties of the language that is being modelled. Thus, a metamodel should be capable of describing a language's syntax, notation and semantics. A *meta-metamodel* is a model that is its own reference model (i.e. conforms to itself) [24]. It is the key to metamodelling as it enables all modelling languages to be described in a unified way, i.e., all metamodels are described by a single meta-metamodel.

Concepts mentioned here represent different tiers of abstractions of the real world, where SUS can be viewed as lowest or tier zero, and meta-metamodel as highest or tier three.

1.2 DSLs vs. GPLs

There is a variety of categories of languages. A distinction is often made between programming languages and modelling languages, but this distinction is currently becoming more and more blurred since programs are treated as models, and some modelling languages may have the executability property. Another distinction is between General Purpose Languages (GPLs) and Domain Specific Languages (DSLs). UML, Java, and C# are examples of GPLs. SQL, HTML, and Excel are examples of DSLs. A DSL is a language designed to be useful for delimited set of tasks, i.e., they have a clearly identified, concrete problem domain, in contrast to GPLs that are supposed to be useful for much more generic tasks, crossing multiple application domains. Domain-Specific Modelling Language (DSML) is a special case of DSL that is used in domain of modelling (as outlined in [28] for Service Modelling).

2 Metamodelling Platforms: An Overview

Metamodelling approaches are an active research field and in the past 20 years serious application areas in the software and information technology industries have been found. Some of them are Enterprise Model Integration (EMI) [8] in the context of Enterprise Application Integration (EAI) [9], Model Integrated Computing (MIC) [10], modelling languages such as the Unified Modelling Language (UML) [11] based on Meta Object Facility (MOF) [12], and model driven development approaches such as Model Driven Architecture (MDA) [13].

Applying the research results of metamodelling approaches metamodelling platforms are developed, like ADO*xx*, industrial software like ADONIS [14], MetaEdit+ [15], modelling frameworks like Eclipse Modelling Framework (EMF) [7], and toolkits like Generic Modelling Environment (GME) [17].

ADO*xx* is an extensible, repository-based metamodelling platform, which offers a three-step modelling hierarchy with a rich meta-metamodel. ADO*xx* can be customized using metamodelling techniques and extended with custom components to build a modelling environment for a particular application domain. ADONIS is a modelling tool based on ADO*xx* for the domain of business process management [14]. The ADO*xx* platform kernel provides basic modules for managing models and metamodels. In addition, the ADO*xx* generic components for graphical and tabular model editing, for model analysis, for simulation, or for model comparison can be reused and customized in all solutions derived from ADO*xx*. Each ADO*xx*-based solution contains a solution-specific modelling language and may have additional set of solution specific components. The scripting language AdoScript provides mechanisms to define specific behavior and functionality. Mechanisms such as *simulation* or *analysis* are defined on meta-meta level and can be redefined on the metamodel level.

MetaEdit+ is a completely integrated environment for building and using individual Domain-Specific Modelling (DSM) solutions [15]. Same as ADO*xx*, it offers a three-step modelling hierarchy. The meta-metamodel forms the *GOPRR* model, offering the basic concepts *Graph*, *Object*, *Property*, *Relationship* and *Role*. A diagram editor, object & graph browsers, and property dialogs support the definition of a new modelling language without manual coding.

OMG's MOF [12], the open source EMF [7] and the Graphical Editor Framework (GEF) [16] are no meta-CASE tools themselves. With MOF the OMG created a meta-metamodel standard, which provides a basis for defining modelling frameworks. UML [11] is an example of instantiated metamodel of the MOF. The EMF which was influenced by MOF is an open source Java based modelling framework and code generation facility for building tools and other applications based on a structured data model. Together with the GEF it provides a possibility to create a new modelling tool.

The GME [17] is a configurable toolkit for creating DSM and program synthesis environments. The configuration is accomplished through metamodels specifying the modelling language of the application domain. The metamodelling language is based on the UML class diagram notation and OCL [18] constraints. The metamodels specifying the modelling language are used to automatically generate the target domain-specific environment. The generated domain-specific environment is then used to build domain models that are stored in a model database or in XML format. GME has a modular, extensible architecture that uses MS COM for integration. GME is easily extensible; external components can be written in any language that supports COM (C++, Visual Basic, C#, Python etc.).

3 Hybrid Modelling

The fundamental integration problem among metamodels (modelling languages) emerges when we try to join together vertically and/or horizontally different metamodels. Metamodels are (i) vertically different, when they vary in the level of details they describe, (ii) horizontally different, when their concepts on the same abstraction level describe different aspects of the system or the same aspect in a different way and (iii) both vertically and horizontally different, when they show characteristics of the previous two. No matter what kind of integration orientation is considered, there is a need to overcome *syntactical*, *structural* and *semantic* discrepancy of metamodels, in order to join their concepts together [25].

Syntactical heterogeneity [25] represents the difference in formats intended for the serialization of metamodels. Two metamodelling platforms can base their serialization mechanisms on different proprietary formats or even paradigms, e.g. having diverse relational, object oriented or XML based schemas.

Structural heterogeneity [25] can be expressed through representational and schematic heterogeneity. Metamodels are represented using different metamodelling languages, i.e. meta-metamodels, each of them showing difference in its expressive power of available modelling primitives (classes, attributes, supported relationship types, etc.). Even when agreed on the common meta-metamodel, metamodels vary schematically when the same concepts being described are modelled in a different way (thus having different conceptual schemas). There are two primary reasons for schematic conflicts: equal concepts are modelled either with different modelling primitives or with different number of primitives.

Semantic heterogeneity [25] includes differences in the meaning of the considered metamodel concepts. Concepts coming from different metamodels can use the same linguistic terms to describe different concepts or use different terms to describe the same concept etc.

A modelling method consists of two components: a modelling technique, which is divided in a modelling language and a modelling procedure, and mechanisms & algorithms working on the models described by the modelling language (see Figure 1). The modelling language contains the elements with which a model can be described. A modelling language itself is described by its syntax, semantics, and notation. The modelling procedure describes the steps applying the modelling language to create results, i.e., models [2]. The amount of requirements concerning defined syntactical rules and modelling steps is influenced by the automated processing that is planned on the created models. This processing is done with the help of mechanisms & algorithms that provide "*functionality to use and evaluate*" models. Basically, when such functionalities, enabling structural analysis (e.g. queries that return activities that meet some defined criteria like costs, delivery times) as well as simulation of models (e.g. prediction of cycle times or staff requirements) are defined for existing modelling techniques, the modelling methods are formed [1].

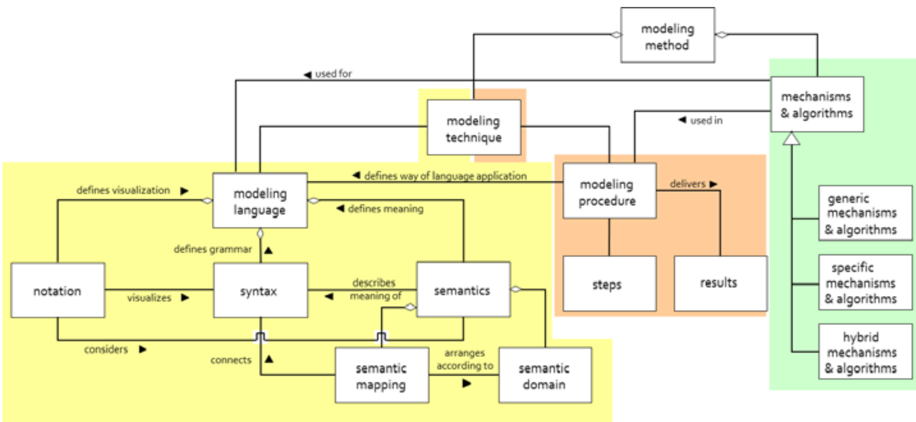


Fig. 1. Modelling methods, mechanisms and algorithms (Karagiannis & Kühn, 2002)

The issue of merging two or more modelling methods (as outlined in [27]) into one can be addressed as *Hybrid Method Engineering* – a combined modelling approach, based on meta-modelling that takes into consideration the different perspectives of modelling languages (metamodels), and results in a comprehensive modelling framework. For this issue to be solved appropriately we need to go through several steps: (i) *integration of modelling methods*, (ii) *support of standards*, and (iii) *merging of different modelling concepts*. In the integration step we need to make sure that different modelling methods, or method chunks are integrable on a common platform. If modelling languages (metamodels) that are a part of modelling methods can be integrated, modelling methods can be integrated as well. In the *support of standards* step we need to check if available standards for modelling aspects are supported, i.e., compliance of models and metamodels to a standard, e.g. ITIL compliance validation. In the last step, *merging of different modelling concepts*, prerequisite is that the platform supports a combined view on concepts from different disciplines [3].

To support the concepts mentioned in this chapter, metamodelling platforms should be realized on a component-based, distributable, and scalable architecture [2]. An important element of metamodelling platform architecture is the meta-metamodel [5]. The meta-metamodel [2] defines general concepts available for method definition and method application such as *metamodel*, *classes*, *relations*, *attributes*, *model types*, etc.

According to Karagiannis & Kühn, other important architectural elements of the metamodelling platform are: *metamodel base*, *model base*, *mechanism base*, *persistency services*, and *access services*. Metamodel base, model base and mechanism base are all based on meta-metamodel and they store, respectively, metamodels, models, and mechanisms. Persistency services support the durable storage of the various bases. These services abstract from concrete storage techniques and permit filling of modelling information in heterogeneous databases, file systems, web services, etc. Access services provide the open, bidirectional exchange of all metamodelling information with other systems, and cover all aspects concerning security such as access rights, authorization, en-/decryption, etc.

A strong model repository is composed of the metamodel, model & mechanism base, persistency services, access services, version control, and validation & verification mechanisms. Furthermore, the model repository needs to be designed to accommodate the reuse of already developed modelling method constructs, and to support pruning and slicing algorithms. If there are such prerequisites, hybrid modelling methods can be easily developed using chunks and pieces from the repository by binding them together into a new coherent whole using appropriate mapping and integration rules.

4 MCG vs. MAS Metamodelling Platforms

Every metamodelling platform is built around a concrete meta-metamodel. This can be a custom meta-metamodel, specifically developed for that platform, or it can be already specified meta-metamodel like MOF. Because most of the modelling platforms have similar foundation, i.e., meta-metamodel, we need to find other key characteristics for metamodelling platform comparison.

Generally, we can divide metamodelling platforms in two groups: the ones that specialize in *model based code generation* (MCG), and the ones that specialize in *model analysis & simulation* (MAS). The other comparison can be based on *value added* to the metamodelling platforms, that is, extra features that are distinguishing one platform from the other.

Most of the metamodelling platforms have additional features that are distinguishing them from other metamodelling platforms. After doing research on most popular metamodelling tools, including ADO*xx*, MetaEdit+, GME, and EMF, a list of important features was compiled (see Table 1). The most influential factors for defining this list of features are: (i) productivity (or rather rise in productivity), (ii) usefulness, and (iii) quality of the modelling tool produced.

Model based code generation (MCG) metamodelling platforms support Model-Driven Engineering (MDE) methodology. MDE is a software development methodology focused on creating and exploiting domain models. In MDE, we use models as the primary artifacts in the development process – we have source models instead of source code [20]. MDE raises the level of abstraction and hides complexity. Truly MDE uses automated transformations in a manner similar to the way a pure coding approach uses compilers. Once models are created, target code can be generated and then compiled or interpreted for execution. From a modeler's

Table 1. List of features used to compare metamodelling platforms

Feature	Description
Language Definition Approach	Graphical, form-based, hybrid
Specifying Notation	Graphical, text-based, hybrid
Syntax Highlighting & Debugging	Support for highlighting, autocompleting, debugging
Scripting	Scripting support for advance customizing
Import & Export	Import & export of metamodels and models
Integration with Other Tools	APIs, Client-Side, Server-Side Integration
Rich Notation	More than simple symbols for nodes and arcs
Dynamic Symbol Change	Symbols change dynamically when model data changes
Different Modelling Views	Modelling, matrix, tabular view

perspective, generated code is complete and it does not need to be modified after generation. For this approach to work, knowledge is not just in the models, but in the code generator and underlying framework. To raise the level of abstraction in MDE, both the modelling language and the generator need to be domain-specific, that is, restricted to developing only certain kind of applications. Focusing on a narrow area of interest makes it possible to map a language closer to the actual problem and makes full code generation realistic – sometimes that is difficult, if not impossible, to achieve with general-purpose modelling languages (UML, etc.).

Model analysis & simulation (MAS) metamodelling platforms support Enterprise Modelling (EM) methodology, including areas like Enterprise Model Integration (EMI) and Enterprise Application Integration (EAI). EM is the abstract representation, description and definition of the structure, processes, information and resources of an identifiable business, government body, or other large organization [22]. It deals with the process of understanding an enterprise business and improving its performance through creation of enterprise models. This includes the modelling of the relevant business domain, business processes, and information technology. In BPM (Business Process Management) models are primarily used for analysis and simulation of the business processes, to find the means to improve their efficiency and quality. Another example of using enterprise models is sharing of knowledge between two or multiple parties (people, departments, companies, etc.). Because, enterprise models are primarily used for analysis & simulation, MAS metamodelling platforms are specialized for creating modelling methods, which are an upgrade on modelling languages, including modelling procedures, algorithms & mechanism (see Figure 1).

5 The Open Model Initiative

The Open Model Initiative (OMI) is an international scientific community, which focuses on the creation, design, evolution and processing of modelling methods and the models designed with them. The initiative is open-membership for all interested experts and organizations, and every *‘model’* which is considered to be useful for a specific purpose by any application domain. The results are public.

OMI provides value both through the modelling and meta-modelling compiler ADO*xx* and also through the social and collaborative platform, by providing knowledge and communities of practice for the development of modelling methods and tools. Adjacent services like OM-TV, the OM-Repository, OM-Apps, OMIverse and OMIpedia give additional features to the initiative.

OMI is structured through its activities in three pillars:

- Community: where groups of individuals share common values and follow common goals. Organized in communities of practice for different domains, they provide value through competence, joint activities, shared practices and resources, sustained interaction, experiences and tools.
- Projects; which can be either (a) modelling projects, thus creating model content for various domains and/or purposes (www.wikimodels.org) and (b) method engineering projects, where the conceptualization of new and further development of existing methods, development and deployment of IT-based modelling tools is realized (www.wikimethods.org), and
- Foundations: which provide modelling languages and algorithms for the processing of models as well as IT-based modelling environments. Additionally this pillar supports designers to choose the right algorithms for the processing of methods and models.

In other words, the OMI platform is a social computing platform which supports the communities through multiple different channels of communication, e.g., forums, blogs, wikis, etc. This model, as mentioned in [23] supports online human interaction and information flow so that communities are formed for ongoing collaboration and exchange of information and knowledge among their members.

6 Conclusion

A metamodel as an idea is introduced to raise the level of abstraction and to simplify the development of modelling languages, modelling methods, and finally, modelling tools. Raising the level of abstraction means making the metamodel flexible, that is, customizable by the metamodelling platform user. This user is sometimes called language engineer. The advantage of using flexible metamodels is direct mapping to the domain under study. There are also benefits that come in form of considerable savings in time and costs and increased quality of delivered solutions.

Due to rapid changing business requirements such as faster time to market, shorter lifecycles, increased interdependencies between business partners, and tighter integration of the underlying information systems, the complexity in developing applications which deliver business solutions is continually growing [2]. This is the reason why the elements of an enterprise are managed more and more model-based, and why are metamodelling platforms getting an integral part of business engineering strategies and approaches. Well-known examples are international standards UML [11] and MOF [12], and metamodelling platforms like ADO*xx* and MetaEdit+ [15]. Additionally, domain specific languages, model transformation approaches, and lifecycle management within large model bases are active research issues.

The other more specific research issue that needs attention is language engineering. Languages are hard to design. The effort that goes into producing a language definition can be overwhelming, particularly if the language is large or semantically rich. That is why reusability is very important. By reusing, rather than re-inventing, it is possible to significantly reduce the time spent on development, allowing language designers to concentrate on the novel features of the language [19].

One of the less noticeable problems is that almost every modelling and metamodelling platform wants to do everything. Most underlying frameworks are very general. Additional functionality for a specific domain of application should be engineered upon the meta-metamodel of the metamodelling platform. That way a new generation of more specialized metamodelling platforms can be developed.

References

1. Karagiannis, D., Grossmann, W., Höfferer, P.: Open Model Initiative A Feasibility Study. University of Vienna, Department of Knowledge Engineering (September 2008)
2. Karagiannis, D., Kühn, H.: Metamodelling platforms. In: Bauknecht, K., Tjoa, A.M., Quirchmayr, G. (eds.) EC-Web 2002. LNCS, vol. 2455, p. 182. Springer, Heidelberg (2002)
3. Xu, T., Ma, W., Liu, L., Karagiannis, D.: Hybrid Modelling: Strategic Model and Business Processes in Active-i*. In: WGBP 2010, Vitoria, ES, Brazil (2010)
4. Brumar, B.A., Popa, E.M.: Advanced techniques for metamodelling. In: Proceeding ICCOMP 2007 Proceedings of the 11th WSEAS International Conference on Computers World Scientific and Engineering Academy and Society (WSEAS) Stevens Point, Wisconsin, USA (2007)
5. Kühn, H., Murzek, M.: Interoperability Issues in Metamodelling Platforms. In: Proceedings of the 1st International Conference on Interoperability of Enterprise Software and Applications (INTEROP-ESA 2005), Geneva, Switzerland. Springer, Heidelberg (2006)
6. Open Model Initiative, `http://www.openmodels.at/`
7. Eclipse Modelling Framework Project (EMF), `http://www.eclipse.org/modelling/emf/`
8. Kühn, H., Bayer, F., Junginger, S., Karagiannis, D.: Enterprise Model Integration. In: Bauknecht, K., Tjoa, A.M., Quirchmayr, G. (eds.) EC-Web 2003. LNCS, vol. 2738, pp. 379–392. Springer, Heidelberg (2003)
9. Linthicum, D.S.: Enterprise Application Integration. Addison-Wesley, Reading (2000)
10. Ledeczi, A., Maroti, M., Bakay, A., Karsai, G., Garrett, J., Thomason IV, C., Nordstrom, G., Sprinkle, J., Volgyesi, P.: The Generic Modelling Environment. In: Workshop on Intelligent Signal Processing at WISP 2001, Budapest, Hungary, May 17 (2001)
11. Object Management Group: Unified Modelling Language, `http://www.omg.org/spec/UML/`
12. Object Management Group: Meta Object Facility, `http://www.omg.org/mof/`
13. Object Management Group: Model Driven Architecture, `http://www.omg.org/mda/`
14. ADONIS, `http://www.boc-group.com/products/adonis/`
15. MetaEdit+, `http://www.metacase.com/`
16. Graphical Editing Framework, `http://www.eclipse.org/gef/`
17. Generic Modelling Environment (GME), `http://www.isis.vanderbilt.edu/projects/GME`
18. Object Constraint Language (OCL), `http://www.omg.org/spec/OCL`

19. Clark, T., Evans, A., Sammut, P., Willans, J.: Applied Metamodelling - A Foundation for Language Driven Development, Version 0.1 (2004)
20. Kelly, S., Tolvanen, J.-P.: Domain-Specific Modelling - Enabling Full Code Generation. Wiley-IEEE Computer Society Press (2008)
21. Frank, U.: Multi-Perspective Enterprise Modelling (MEMO) - Conceptual Framework and Modelling Languages. In: Proceedings of the 35th Hawaii International Conference on System Sciences, HICSS 2002 (2002)
22. Fox, M.S., Gruninger, M.: Enterprise Modelling. AI Magazine 19(3) (1998)
23. Capuruço, R.A.C., Capretz, L.F.: A Unifying Framework for Building Social Computing Applications. In: Lytras, M.D., Damiani, E., Tennyson, R.D. (eds.) WSKS 2008. LNCS (LNAI), vol. 5288, pp. 11–21. Springer, Heidelberg (2008)
24. Kurtev, I., Bezivin, J., Jouault, F., Valduriez, P.: Model-based DSL Frameworks. In: Companion to the 21st ACM SIGPLAN Conference on Object-Oriented Programming Systems, Languages, and Applications (OOPSLA 2006), Portland, Oregon, USA (2006)
25. Zivkovic, S., Kühn, H., Karagiannis, D.: Facilitate Modelling Using Method Integration: An Approach Using Mappings and Integration Rules. In: ECIS 2007 Proceedings. Paper 122 (2007)
26. Moody, D.L.: Theoretical and practical issues in evaluating the quality of conceptual models: current state and future directions. Data & Knowledge Engineering 55, 243–276 (2005)
27. Utz, W., Woitsch, R., Karagiannis, D.: Conceptualisation of Hybrid Service Models: An Open Models Approach. In: The 4th International IEEE Workshop on Service Science and Systems, Held in Conjunction with COMPSAC 2011 (in press, 2011)
28. Hrgovcic, V., Utz, W., Karagiannis, D.: Service Modeling: A Model Based Approach for Business and IT Alignment. In: The 5th International IEEE Workshop on Requirements Engineering for Services, held in Conjunction with COMPSAC 2011 (in press, 2011)

Using Semantically Annotated Models for Supporting Business Process Benchmarking

Hans-Georg Fill

Stanford University BMIR / University of Vienna DKE,
251 Campus Drive, Stanford CA 94305, USA
fill@dke.univie.ac.at

Abstract. In this paper we describe an approach for using semantic annotations of process models to support business process benchmarking. We show how semantic annotations can support the preparation of process benchmarking data by adding machine-processable semantic information to existing process models without modifying the original modeling language, conduct semantic analyses for the purpose of performance measurement, and obfuscate the information contained in the models for ensuring confidentiality. The approach has been implemented on the ADOxx platform and applied to two use cases for a first evaluation.

Keywords: Semantic Annotation, Benchmarking, Performance Evaluation, Ontologies.

1 Introduction

Evaluating the performance of business processes and comparing it to internal and external benchmarks is an essential aspect in performance management [1]. The core feature of such benchmarks is to learn from others and then adapt one's own processes in order to gain competitiveness [2]. In this way, large, global companies aim today for the spreading and homogenization of their internal best practices across their units [3]. However, as processes are today also viewed as assets that represent the actual know-how platform of an organization and are thus essential for the creation of competitive advantage [4,5], benchmarking should also allow for the combination with individual, innovative solutions [6].

As a basis for business process benchmarking it can be reverted to graphical modeling languages [7]. These can act as a foundation for process benchmarking by providing a machine-processable representation that can be easily shared between interested parties. However, as benchmarking may require additional or different information than was originally conceived when creating the models, additional manual effort is needed to prepare suitable models [8]. This concerns especially the requirements of comparability and confidentiality as well as the content-based analysis of the models.

To support users in preparing such data for analyses and benchmarking, we will describe an approach that builds upon semantically annotated process models. Thereby, the preparation and analysis of the process data is supported in the

J. Grabis and M. Kirikova (Eds.): BIR 2011, LNBIP 90, pp. 29–43, 2011.

following ways: Through annotations with terms from a shared vocabulary the comparability of individually created process models can be ensured. In addition, we will show how semantic annotations enable ex-post analyses of the semantic content of the process models to semi-automate complex analysis tasks. Furthermore, the annotations create the basis for an obfuscation mechanism that deals with sensitive information and acts as an additional incentive to share the data. The remainder of the paper is structured as follows: In section 2 we will outline the foundations for our approach which will allow us to describe the approach in section 3. Section 4 shows the implementation and application to two uses cases, section 5 discusses the relations to existing approaches and section 6 will conclude the paper with an outlook on the future steps.

2 Foundations

In this section we will give a brief outline of the foundations for our approach. In particular, we will describe the aspects of business process benchmarking, the role of conceptual modeling for supporting benchmarking, and the characteristics of semantic annotations of conceptual models that are relevant for our approach.

2.1 Business Process Benchmarking

Benchmarking in general refers to the continuous measurement and comparison of an organization against other organizations, in particular business leaders, to get information helping to improve its performance [2,1,8]. Thereby, it is aimed at establishing objective measures of an organization's performance, the adaptation of best practices, and the incentive for introducing new and innovative concepts [6]. Traditionally, a distinction has been made between internal benchmarking, competitive benchmarking, and generic benchmarking [2,9]. Internal benchmarking is particularly used in large organizations and characterizes comparisons against other units in the same organization. In competitive benchmarking the own performance of an organization is compared to its direct competitors and in generic benchmarking the comparison is performed regardless of industry. Especially in competitive benchmarking, confidentiality and sensitivity of data and information may pose potential problems. This concerns in particular *business process benchmarking*, which measures, compares, and exchanges the practices and ways of performing and not only the pure levels of performance [2]. Business process benchmarking is therefore able to give deeper insights into the capability and choices for improving an organization's performance.

Typically, business process benchmarking is comprised of four steps: The planning of the benchmarking project, the collection of process data, the analysis of the data for results, and the adaptation for improvement [1]. In the following we will focus on the collection of process data and the analysis of the data. In this way, at first the business processes to be benchmarked are selected and key process performance indicators are determined. Due to the large number of existing metrics we refer to a recent survey by Heinrich and Paech who compiled quality characteristics for process activities, actors, and inputs and outputs [10]. The comparison of these indicators then shows how much process improvement is possible in relation to the benchmarking partner, whereas the comparison of qualitative process information, i.e. the process flow, gives detailed insights on where and how improvements can be achieved [9].

2.2 Conceptual Process Models for Benchmarking

As mentioned in the introduction, organizations today often represent the knowledge about their business processes in the form of conceptual graphical models [7]. These are based on a modeling language with a formal syntax and a graphical notation that can serve as a basis for the implementation of according model editors [11]. One particular aspect - which we will show is of major importance for a machine-based support of benchmarking - is that the semantics of the labels in conceptual models are not formally defined but given in natural language [12]. This stems from the goal of conceptual modeling to support human communication and understanding, which does not require a formal representation of the meaning [13]. Although this greatly eases the understanding and handling of these types of modeling languages and has contributed to their widespread use [14], it also limits the options for processing the contained information with algorithms.

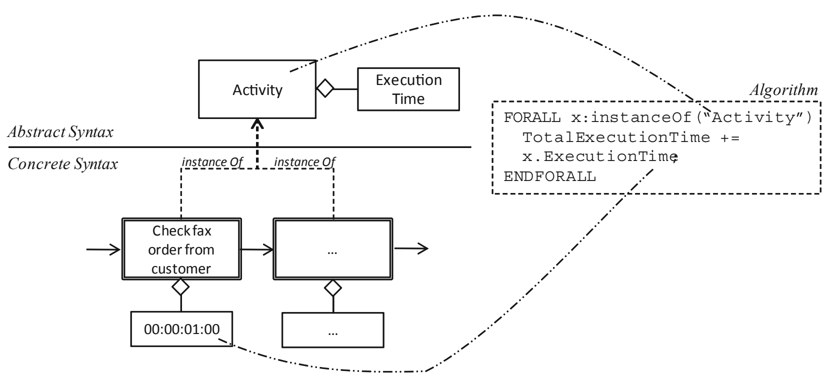


Fig. 1. Illustration of an Algorithm based on the Abstract and Concrete Syntax of a Business Process Model

For illustrating this with an example consider an abstract syntax element with the label "activity" (see also figure 1). To correctly interpret this element a semantic description has to be supplied for it. Thereby, it can be defined that elements of this type consume time, whose quantity is specified by the attached attribute "execution time". Among the many ways of expressing such semantic descriptions, one approach is to create an algorithm that explicitly references elements of the abstract syntax and can thus also process their instantiations. In this way semantic definitions may be specified independently of the instantiations of the element as they are targeted towards the abstract syntax that is the same for all resulting models.

However, when an element is instantiated, a user typically adds a label and a description to the element. Following the example, a label for the activity element could be “Check fax order from customer”, together with a textual description that explains in detail which aspects should be checked. Without further specification, the semantic information contained in these natural language descriptions cannot be directly processed by an algorithm as no formal interpretation has been defined. In contrast to the specification of formal semantics for the abstract syntax, this semantic information entirely depends on the choice of the user at the time of creating a model and is not a-priori known.

When applying these considerations to benchmarking, an essential aspect is the computation of selected key performance indicators based on the process data. In particular, several types of indicators can already be derived by accessing the structural properties of the process models via their abstract syntax. This includes static metrics such as the number of events, activities or decisions as well as flow metrics such as the number of loops, parallel paths, joins or splits [15]. However, a large part of quality metrics require the interpretation of the content of the business processes. Examples for such metrics include but are not limited to the number of media disruptions, the fault density, the degree of automation or the effectiveness of documentation [10].

Depending on the type of the used modeling language, this information may however not be at all or only partly expressible by the abstract syntax. It may be at best added by the user through the labels in natural language at the time of modeling. And even more, it could also be the case that the labels alone do not reflect this information at all, but that it actually has to be added by the user specifically for the purpose of benchmarking. When aiming for an algorithmic support of benchmarking, it is though necessary to lift this information to a concept-based level where this information is provided in a machine-processable format.

2.3 Semantic Annotation of Conceptual Process Models

In order to explicate such semantic information it can be chosen from several directions. One is to create a *new modeling language*, either from scratch or by extending an existing one. This allows defining or extending the abstract syntax in a way that the parts of the semantic information, which are necessary for a particular algorithm can be expressed. It also implies however, that the modeling language and according models can be adapted in this way. Typically, this is accomplished by introducing new elements or attributes that provide the necessary information. Thereby, each instance can be assigned the semantic information required for running the algorithms. Examples for this direction are domain specific languages in general, the semantic building block-based language [16] and the visual templates described by [17]. Although such approaches provide many advantages in terms of machine-processing, their major drawback is that the modeler may not be able to represent all relevant world facts but is limited to the terms and concepts available in the extended abstract syntax [16].

Another direction is to use an *existing modeling language* and ensure the processability either by enforcing modeling conventions or by using references to machine-processable semantic schemata, i.e. *semantic annotations*. Modeling conventions may either be defined on an organizational level [18], e.g. by providing rules which terms can be used in a model. Or, they may be enforced automatically, e.g. by applying natural language processing techniques together with a domain vocabulary [19]. The use of semantic annotations has been described by several authors. Thereby, the references or meta-data can be expressed using different types of languages, ranging from conceptual languages that are based on natural language semantics to programming languages and logic based languages with formal semantic specifications. This direction has been successfully applied to several common business process modeling languages such as: Event-driven process chains [20,21], the business process modeling notation [22] or Petri nets [23]. For these approaches the required semantic information is either derived by automatically looking up terms in a schema or linking them to the schema manually and then building on this information for further processing. In the following we will use this last direction for illustrating how semantic annotations can be used for supporting the data preparation and analysis step in business process benchmarking.

3 Model-Based Benchmarking Using Semantic Annotations

To discuss our approach we will revert to meta models for describing the abstract syntax of the underlying modeling languages [11]. This will permit us to illustrate the relationships between business process models and semantic schemata in an intuitive way. Furthermore, by using meta models it will be possible to directly implement the modeling language on a corresponding meta modeling platform as well as specify algorithms for computing process performance indicators.

3.1 Definition of the Meta Model

We first define a meta model for describing the abstract syntax of a simplified business process model type (see figure 2). The meta model comprises elements for describing the control flow and information flow in business processes as well as the organizational structure. In contrast to real-world business process modeling languages both model types only contain a subset of possible modeling elements and relations. However, for the purposes of illustrating the concepts necessary for our approach this is sufficient and could easily be extended at any time. Similarly, also the attributes assigned to the model elements have been limited to the "name" attribute for process elements, the "probability" attribute for the sequence flow relation and the "execution time" attribute for the "activity" elements.

For describing machine-processable semantic schemata, we added elements to the meta model for representing concepts of the web ontology language (OWL). OWL has been chosen as it is currently widely used for representing formal semantic information. In particular, OWL builds upon description logics [24] and comes with a formal semantic specification. Thereby, it supports automated reasoning techniques for checking the consistency of an ontology and the entailment relationships between its concepts based on a set of axioms [25]. So it can for example be ensured by a reasoner that the semantic information stays consistent with previously defined concepts based on a detailed set of restrictions. Furthermore, OWL ontologies can be exchanged using an RDF/XML syntax and thus may be easily shared between different tools and platforms. In figure 2 the main elements of OWL are shown. To define semantic annotations of conceptual models, two options have been made available in the meta model. The first is by using the reference attribute "semantic annotation" that has been added to the "Process Element" and the "Organizational Element" super classes.

By using this attribute, semantic annotations can be easily expressed that result in a direct assignment of ontology concepts to instances of the concrete syntax in the business process and working environment models. However, for this kind of annotation it is necessary to extend the abstract syntax with the annotation attribute. Therefore, also a second option has been included: By using the separate "Semantic Annotation Model", the annotations can also be defined without modifying the underlying business process language and the annotation is stored separately from the conceptual models and ontology models. This uncoupling of conceptual models, annotations, and ontology models also provides a further advantage in terms of flexibility: By technically separating the annotations from the original models, the annotations do not affect the original process models and can be treated independently.

3.2 Semantic Annotations for Business Process Benchmarking

In the following we describe three particular aspects of using semantic annotations in business process benchmarking: For the *annotation* of business process and organizational models during data preparation, for the *analysis* of process data based on these annotations, and for the *semantic obfuscation* of data.

To show how semantic annotations can be used for benchmarking, we will use the example of determining the number of media disruptions in a particular existing process model. It is assumed that an existing process model shall be complemented with semantic annotations in order to integrate the additional knowledge about the occurrence of media disruptions. Therefore, we regard a segment of a business process that contains two instances of activities (see figure 3): The name attribute of the first activity instance is filled with the value "Check fax order from customer" and the name attribute of the second instance with "Enter order information in booking application". It is further assumed that the activities are connected by the "Sequence Flow" relation, which is shown in figure 3 by an arrow between the two instances.

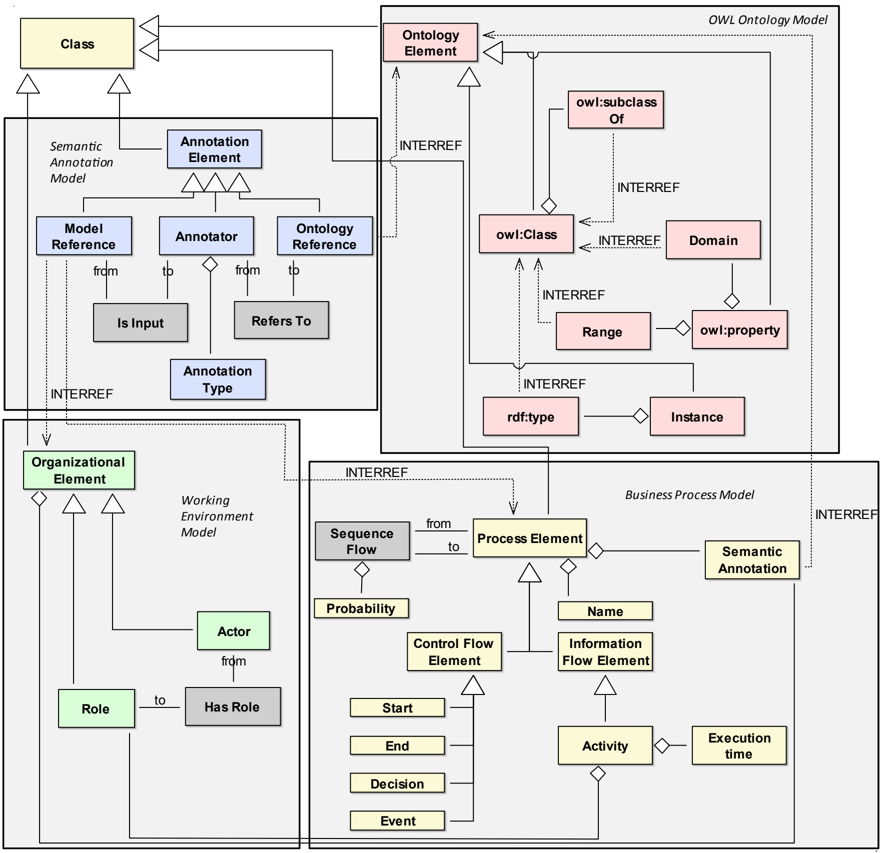
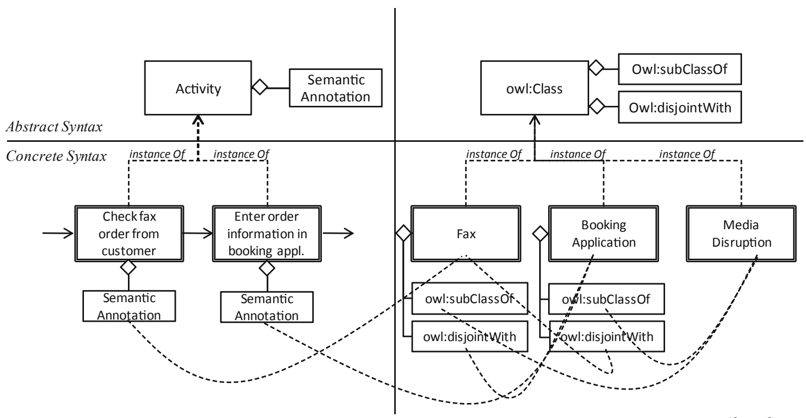


Fig. 2. Excerpt of the Meta Model for Semantic Annotations of Business Processes

At the same time, an ontology model is defined that contains three instances of "owl:Class". Thereby, ontology concepts are defined with the following URIs - for enhancing readability we leave out the preceding namespace definitions: "Media Disruption", "Booking Application", and "Fax". "Fax" and "Booking Application" are defined as sub classes of "Media Disruption" by adding the reference in the "owl:subClassOf" attribute. Due to the open world assumption used by OWL, the two sub classes of "Media Disruption" have to be explicitly defined as being disjoint. Therefore, the "owl:disjointWith" attributes of both sub classes are filled with a reference to the respective other sub class. The actual annotation of the activity instances can now be accomplished by adding the according references to the "Semantic Annotation" attributes.

With these definitions in place we can now outline how an algorithm can be implemented that computes the number of media disruptions in a process model (see figure 3). The depicted algorithm assumes that the process only consists of a sequence of activity instances that are stored in the array "x[]". It then iterates over this sequence of activities and determines whether there is a change in the semantic annotations that have to be sub classes of "Media Disruption" and disjoint from the next annotation. Until here, the use of OWL for describing the ontology has offered some advantages by providing useful axioms such as subClassOf and disjointWith. However, the formal semantics available for OWL provide additional options. Suppose one would like to add more specific types of booking applications. In this case the ontology could be easily extended by adding sub classes to "Booking Application". An automatic OWL reasoner can then check if the ontology is consistent with the previously defined concepts, e.g. that only disjoint concepts are used. At the same time, the outlined algorithm is still applicable based on another result by the reasoner: As sub class relationships in OWL are transitive, it can be inferred that also all sub classes of "Booking Applications" are sub classes of "Media Disruption".



Algorithm

```
SET i=0
SET x[]:instanceOf("Activity")
WHILE x[i].hasNext
    IF (x[i].SemanticAnnotation.subClassOf("MediaDisruption") &&
        x[i+1].SemanticAnnotation.subClassOf("MediaDisruption")) &&
        x[i].SemanticAnnotation.disjointWith(x[i+1].SemanticAnnotation)
    THEN
        SET MediaDisruption++;
    ENDIF
    i++;
ENDWHILE
```

Fig. 3. Illustration of Computations based on Annotated Models

The hierarchical character of OWL ontologies can also serve another requirement in benchmarking: As an ontology typically contains general concepts that are specialized into more specific ones, this information can be used to abstract information. This can be applied for meeting the requirements of confidentiality when sharing process data for benchmarking purposes. As shown in figure 4, the annotation of a process element (Step 1) allows the semantic abstraction to more general concepts (Step 2) which can then be assigned as a new name for the corresponding element (Step 3).

We denote this type of abstraction as *semantic obfuscation* because it does not completely remove the semantic information. It therefore still allows conducting semantic analyses of the underlying processes at least to a certain level. Furthermore, the reference to the original process models can still be preserved. In this way an external evaluator can analyze the obfuscated models and run algorithms on them while still being able to give feedback on where particular process parts may need to be improved. The example in figure 4 shows an excerpt of the account opening process at a bank. By annotating the process elements with concepts from an ontology (Step 1), the information can be abstracted based on the hierarchy defined in the ontology (Step 2), and the new information assigned to the corresponding model element (Step 3). Depending on the degree of obfuscation the user chooses, the corresponding higher level ontology concept is used as a replacement for the name attribute. Although the semantic information is then only available in an abstracted form, the remaining process information such as execution times or transition probabilities are fully preserved.

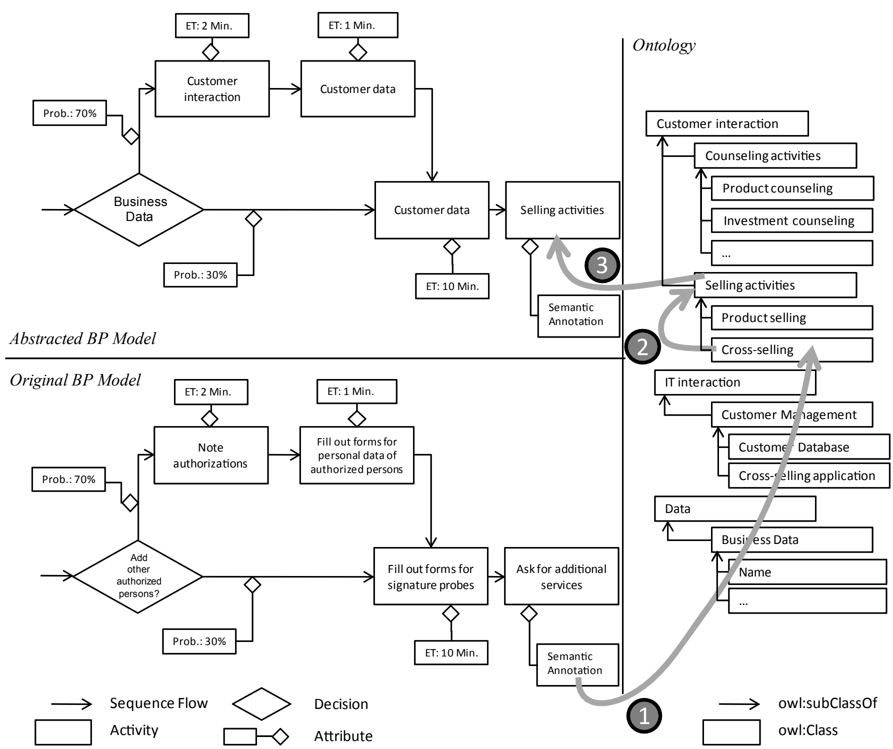


Fig. 4. Illustration of Semantic Obfuscation for Business Process Models

4 Implementation and Application

We have implemented a model editor for the described meta model by using the ADOxx meta modeling platform, which provides the scripting language ADOscript, the generic model query language AQL as well as import and export functionalities for exchanging models [26][1]. ADOxx was chosen primarily based on its industry-ready scalability and existing knowledge in regard to its implementation languages on the side of the authors. Furthermore, to support the semantic obfuscation of models we have implemented an algorithm in ADOscript that allows the automatic abstraction of the name attributes in the business process models by accessing the "owl:subClass" information provided in the OWL models. Thus, the algorithm is able to abstract each referenced concept by stepping up the class hierarchy and then assigning the name of the upper concept together with a unique ID as a label to the original element. The algorithm currently only supports single inheritance relationships, which proved however sufficient so far during the first evaluation.

In order to integrate OWL ontologies, a coupling between the Protégé ontology toolkit[2] and the ADOxx platform was established. The exchange of ontology information is thereby realized via a plugin for Protégé that translates the ontology information into the generic XML format of the ADOxx platform. In this way, the generic AQL query language can be used to retrieve information about the semantic annotations and the according ontology models. At this stage the resulting tool can be used for creating business process and working environment models and annotate them with the concepts from the imported OWL ontologies. By using the AQL query language, models can then be analyzed and compared.

For a first evaluation we applied the tool to two scenarios: The first one is based on the benchmarking of service interaction processes for the Bulgarian and Romanian chambers of commerce that have been previously elaborated in the LD-CAST project[3] (see figure 5). To enable the semantic annotation of these processes, a specific benchmarking ontology was first developed in Protégé. It provides a simplified description of the domain of business process benchmarking and contains in particular the OWL classes "Automated_business_process_task", "Business_process_task_with_media_disruption", and "Manual_business_process_task". The ontology was then translated into the ADOxx XML format and made available as an OWL model. The annotation of the service processes could be easily accomplished by linking the activity elements to the corresponding ontology concepts. By using the AQL query language, the statistics for the benchmarking in regard to the degree of automation and the occurrence of media disruptions could then be successfully retrieved. As an example the syntax of an

[1] The implementation will be made freely available for further evaluation in the course of the SeMFIS project of the Open Model Initiative at http://www.openmodels.at/web/semfis

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

[3] LD-CAST stands for Local Development Cooperation Actions Enabled By Semantic Technology, http://www.ldcastproject.com

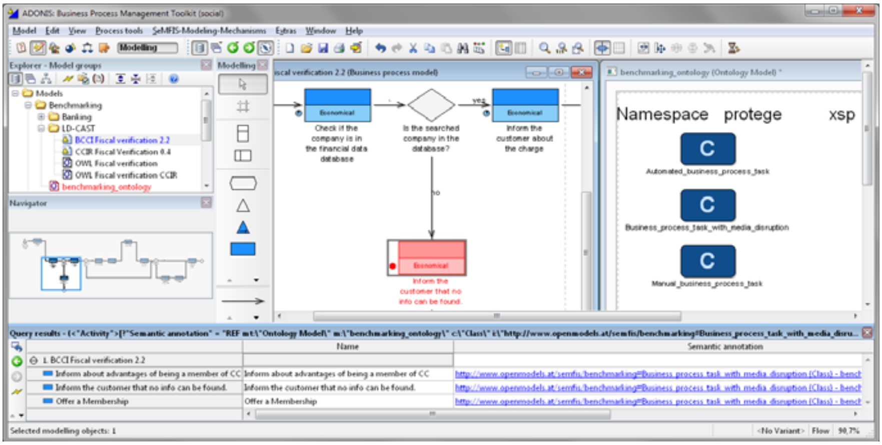


Fig. 5. Screenshot of the ADOxx Implementation

AQL query for retrieving Activity elements from a Business Process Model that have been annotated with the OWL Class "Business_process_task_with_media_disruption" is given in the following:

Sample AQL Query:

```
(<"Activity">
[?"Semantic annotation" =
  "REF mt:\"Ontology Model\"
   m:\"benchmarking_ontology\"
   c:\"Class\"
   i:\"http://www.openmodels.at/semfis/benchmarking#
   Business_process_task_with_media_disruption\""])
```

The second scenario comprised the application of the semantic obfuscation approach to the banking domain. Based on two processes for the opening of accounts at two Swiss banks that are publicly available and have been further refined by the author based on their documentation[4], the obfuscation algorithm could be applied. For this purpose, a simplified domain ontology for banking was elaborated in Protégé. This ontology comprised in particular several subclass and superclass relationships, e.g. by using a general concept such as "Selling_activites" and a specialized concept "Cross-selling_activity". In comparison to the first scenario, the semantic annotation in this case required more effort as all necessary domain concepts had to be assigned to the activities to accurately describe the content. After the annotation the obfuscation algorithm could be successfully applied which led to the outcomes shown in figure 6.

[4] The processes have been elaborated and published by Gerardo Palmisano for the Hypothekarbank Lenzburg and by Jonas Winkler for the Spar- und Leihkasse Riggisberg on http://www.lernender.ch

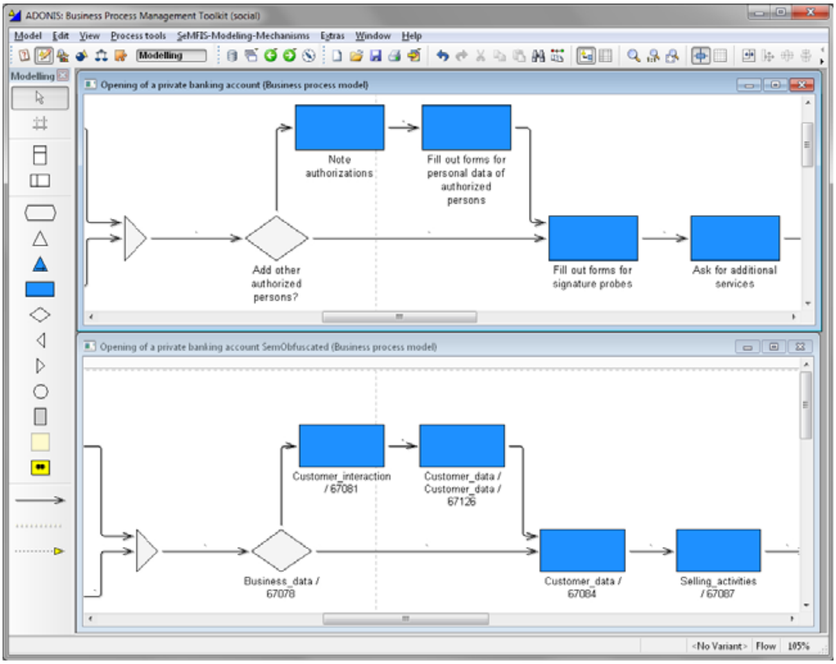


Fig. 6. Example for the Semantic Obfuscation Using an Account Opening Process

5 Related Work

When comparing our approach to existing work, three categories of related approaches can be identified: Approaches that deal with IT-based methods for *business process benchmarking*, approaches that describe *the application of semantic annotations of process models and according tools*, and *semantic schemata for process evaluation*. For the first category, the approach by [9] who describe a logic based approach for the comparison of business processes to support benchmarking, is closely related to our approach. However, in contrast to our approach a common data dictionary is assumed to already exist which does not provide the expressiveness, and shareability of OWL ontologies. Similarly, the approaches by [16] who describe a method to construct comparable business process models using a domain specific language and [19] who present a method to create naming conventions for arbitrary conceptual models share several aspects. As already pointed out in section 2.3, our approach does however not enforce a particular modeling language but instead uses additional, decoupled semantic information that is machine-processable.

In regard to approaches that discuss the use of semantically annotated business process models several related approaches exist. These include but are not limited to the detection of regulatory compliance [27,28], semantic reverse business engineering in order to analyze productive ERP systems [12,29], semantic model comparisons [23], to support cross-organizational business processes and interoperability [30] or for the dynamic binding of web services during the execution of business processes [21]. These approaches may be directly used in addition to our approach, e.g. to add performance indicators that measure the compliance of processes, to map the data acquired during the execution of processes into the process models, to identify similar processes prior to benchmarking, to integrate different types of modeling languages or to automatically plan and execute processes based on the insights gained through benchmarking.

Concerning the tool support for semantic annotations of process models several options are available: These include WSMO studio [31] that has been specifically developed in the SUPER-IP project in regard to automated process execution, the semantic extensions for the Maestro BPMN tool [22] and ARIS [21], and SemPeT for realizing semantic annotations for Petri nets [23]. Although these tools may also be used for applying our approach, they were either not available due to their licenses or used specific modeling and ontology languages that are different to the ones we proposed in our approach. However, apart from licensing issues it should be possible to adapt these tools to support our approach as well.

Semantic schemata that can be used for the annotations include not only approaches based on formal semantics such as the ontology proposed for key performance indicators [32], the Core Ontology for Business Process Analysis (COBRA) [33], the OWL based business process management ontology (BPMO) by [34] or the business process ontology (BPO) by [35], but also semi-formal approaches such as the schema for monitoring and analyzing processes by [36]. Furthermore, any kind of domain ontology that contains a hierarchical structure necessary for the semantic obfuscation may be applied. In order to directly use them for our approach they would have to be either available in the OWL format or be translated to it.

6 Conclusion and Outlook

With the proposed approach of using semantic annotations for business process models it could be shown how these additional technical functionalities can help to prepare process data for benchmarking and conduct machine-based semantic process analyses. The approach can be easily extended to other model types, e.g. by accessing the information in the organizational model and obfuscating the names and roles of particular persons that participate in a business process or by additional resource models, e.g. for analyzing the IT usage in a business process. Apart from the functionality presented here, the next step will be to apply the approach in practice and conduct according empirical research and user studies to further evaluate its applicability. Thereby the techniques for accomplishing the annotations shall be further detailed based on the feedback from domain experts. Furthermore, the distribution of model content using the discussed semantic obfuscation technique will have to be further evaluated in terms of maintaining confidentiality for conducting benchmarks in practice.

Acknowledgement. The work on this paper has been funded by the Austrian Science Fund (FWF) in the course of an Erwin-Schrödinger-Fellowship grant number J3028-N23.

References

1. Camp, R.: Business Process Benchmarking: Finding and Implementing Best Practices. ASQC Quality Press, Milwaukee (1995)
2. Andersen, B.: Benchmarking. In: Rolstadas, A. (ed.) Performance Management - A Business Process Benchmarking Approach, pp. 211–242. Chapman and Hall, London (1995)
3. Amaral, P., Sousa, R.: Barriers to internal benchmarking initiatives: An empirical investigation. Benchmarking: An International Journal 16(4), 523–542 (2009)
4. Karagiannis, D., Woitsch, R.: The PROMOTE approach: Modelling Knowledge Management Processes to describe an organisational KMS. In: ECAI Workshop, Lyon, France, INRIA (2002)
5. Hill, J.: Highlights From the Business Process Improvement Track at Symposium/ITxpo, Gartner Research (2010), http://www.gartner.com
6. Legner, C., Oesterle, H.: Process benchmarking - A methodological approach for process development using standard software (in German). In: 4th International Conference on Business Informatics, Physica, Heidelberg (1999)
7. List, B., Korherr, B.: An Evaluation of Conceptual Business Process Modelling Languages. In: SAC 2006. ACM, Dijon (2006)
8. Hinton, M., Francis, G., Holloway, J.: Best practice benchmarking in the UK. Benchmarking: An International Journal 7(1), 52–58 (2000)
9. Juan, Y., Ou-Yang, C.: A Process Logic Comparison Approach to Support Business Process Benchmarking. Int. Journal of Advanced Manufacturing Technologies 26, 191–210 (2005)
10. Heinrich, R., Paech, B.: Defining the Quality of Business Processes. In: Modellierung 2010, pp. 133–148. GI LNI, Klagenfurt (2010)
11. Karagiannis, D., Kühn, H.: Metamodelling platforms. In: Bauknecht, K., Tjoa, A.M., Quirchmayr, G. (eds.) EC-Web 2002. LNCS, vol. 2455, p. 182. Springer, Heidelberg (2002)
12. Celino, I., De Medeiros, A., Zeissler, G., Oppitz, M., Facca, F., Zoeller, S.: Semantic business processes analysis. In: Hepp, M., Hinkelmann, K., Karagiannis, D., Klein, R., Stojanovic, N. (eds.) SBPM 2007 Workshop, CEUR Workshop Proceedings, vol. 251 (2007)
13. Mylopoulos, J.: Conceptual Modeling and Telos. In: Loucopoulos, P., Zicari, R. (eds.) Conceptual Modelling, Databases and CASE: An Integrated View of Information Systems Development, pp. 49–68. Wiley, Chichester (1992)
14. Rosemann, M.: Potential pitfalls of process modeling: Part A. BPM Journal 12(2), 249–254 (2006)
15. Cardoso, J., Mendling, J., Neumann, G., Reijers, H.A.: A discourse on complexity of process models. In: Eder, J., Dustdar, S. (eds.) BPM Workshops 2006. LNCS, vol. 4103, pp. 117–128. Springer, Heidelberg (2006)
16. Becker, J., Breuker, D., Pfeiffer, D., Raeckers, M.: Constructing comparable business process models with domain specific languages - an empirical evaluation. In: ECIS 2009, Verona, Italy (2009)
17. Peleg, M., Tu, S.: Design Patterns for Clinical Guidelines. AI in Medicine 47(1), 1–24 (2009)
18. 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)
19. Becker, J., Delfmann, P., Herwig, S., Lis, L., Stein, A.: Towards Increased Comparability of Conceptual Models - Enforcing Naming Conventions through Domain Thesauri and Linguistic Grammars. In: ECIS 2009, Verona, Italy (2009)

20. Thomas, O., Fellmann, M.: Semantic Business Process Management: Ontology-based Process Modeling Using Event-Driven Process Chains. IBIS 2(1), 29–44 (2007)
21. Stein, S., Stamber, C., El Kharbili, M.: ARIS for Semantic Business Process Management. In: Ardagna, D., Mecella, M., Yang, J. (eds.) Business Process Management Workshops. LNBIP, vol. 17, pp. 498–509. Springer, Heidelberg (2009)
22. Born, M., Hoffmann, J., Kaczmarek, T., Kowalkiewicz, M., Markovic, I., Scicluna, J., Weber, I., Zhou, X.: Semantic Annotation and Composition of Business Processes with Maestro. In: Bechhofer, S., Hauswirth, M., Hoffmann, J., Koubarakis, M. (eds.) ESWC 2008. LNCS, vol. 5021, pp. 772–776. Springer, Heidelberg (2008)
23. Ehrig, M., Koschmider, A., Oberweis, A.: Measuring Similarity between Semantic Business Process Models. In: Roddick, J., Hinze, A. (eds.) APCCM 2007, vol. 67, pp. 71–80. ACM, New York (2007)
24. W3C: Owl web ontology language - overview w3c recommendation (February 10, 2004), http://www.w3.org/TR/owl-features/ (accessed September 16, 2005)
25. Horrocks, I., Patel-Schneider, P., Van Harmelen, F.: From SHIQ and RDF to OWL: The Making of a Web Ontology Language. Web Semantics: Science, Services and Agents on the World Wide Web 1(1), 7–26 (2003)
26. Fill, H.G.: UML Statechart Diagrams on the ADONIS Metamodeling Platform. Electronic Notes in Theoretical Computer Science 127(1), 27–36 (2004)
27. Governatori, G., Hoffmann, J., Sadiq, S., Weber, I.: Detecting Regulatory Compliance for Business Process Models through Semantic Annotations. In: 4th Business Process Design Workshop. Springer, Milan (2008)
28. El Kharbili, M., Stein, S., Markovic, I., Pulvermueller, E.: Towards a Framework for Semantic Business Process Compliance Management. In: GRCIS 2008 Workshop (2008)
29. Oppitz, M., Zeissler, G., Zoeller, S., Carenini, A.: Semantic Reverse Business Engineering - Public Deliverable 5.7 of Project IST 026850 SUPER (2009), http://www.ip-super.org/res/Deliverables/M36/D5.7.pdf (accessed March 15, 2011)
30. Hoefferer, P.: Achieving Business Process Model Interoperability Using Metamodels and Ontologies. In: Oesterle, H., Schelp, J., Winter, R. (eds.) ECIS 2007, pp. 1620–1631. University of St. Gallen, St. Gallen (2007)
31. Dimitrov, M., Simov, A., Momtchev, V., Konstantinov, M.: WSMO Studio – A Semantic Web Services Modelling Environment for WSMO. In: Franconi, E., Kifer, M., May, W. (eds.) ESWC 2007. LNCS, vol. 4519, pp. 749–758. Springer, Heidelberg (2007)
32. Wetzstein, B., Karastoyanova, D., Leymann, F.: Towards Management of SLA-Aware Business Processes Based on Key Performance Indicators. In: 9th Workshop on Business Process Modeling, Development, and Support (2008)
33. Pedrinaci, C., Domingue, J., Alves de Medeiros, A.K.: A core ontology for business process analysis. In: Bechhofer, S., Hauswirth, M., Hoffmann, J., Koubarakis, M. (eds.) ESWC 2008. LNCS, vol. 5021, pp. 49–64. Springer, Heidelberg (2008)
34. Jenz & Partner: The Open Source Business Management Ontology (BMO) (2006), http://www.bpiresearch.com/Resources/RE_OSSOnt/re_ossont.htm (accessed June 13, 2006)
35. Markovic, I., Pereira, A.C.: Towards a formal framework for reuse in business process modeling. In: ter Hofstede, A.H.M., Benatallah, B., Paik, H.-Y. (eds.) BPM Workshops 2007. LNCS, vol. 4928, pp. 484–495. Springer, Heidelberg (2008)
36. Karagiannis, D., Ronaghi, F., Fill, H.G.: Business-oriented IT management: developing e-business applications with E-BPMS. In: ICEC 2008, vol. 258, pp. 97–100. ACM, New York (2007)

Performance Measurement Framework with Formal Indicator Definitions

Aivars Niedritis, Laila Niedrite, and Natalija Kozmina

Faculty of Computing, University of Latvia, Raina boulv.19, Riga
{Aivars.Niedritis,Laila.Niedrite,Natalija.Kozmina}@lu.lv

Abstract. Definition of appropriate measures of organization's performance should be conducted in a systematic way. In this paper the performance measurement and indicators are discussed not only from the side of management models, but also from the point of view of measurement theories to find out appropriate definitions. In our work we propose a formal specification of indicators. The principles of indicator reformulation from free form indicators to formal requirements are formulated and applied in several examples from performance measures database. The formally defined indicators could be used in the proposed performance measurement framework that covers five-step indicator lifecycle.

Keywords: performance measurement, key performance indicators, data warehouse.

1 Introduction

In a long-term perspective it is necessary not only to understand the current situation in an organization and to rebuild the business processes, if necessary, in the most effective way, but also to continue the improvement of business processes based on comprehensive measurement of organization's performance.

Effective organization of business processes ensures the achievement of institution's goals. During a performance measurement, the measurement results should be compared with the target values to make decisions, whether goals are achieved or not. Organizations use performance measures to align daily activities to strategic objectives [1]. The role of appropriate measures could not be underestimated, so Harrington [2] stated: "Measurements are the key. If you cannot measure it, you cannot control it. If you cannot control it, you cannot manage it. If you cannot manage it, you cannot improve it."

An important aspect is how to choose appropriate measures and how to define an appropriate measurement framework. The performance measurement should be performed from different perspectives.

Companies should use performance measurement systems, if they want to succeed in the competition with other companies. Measurement systems have to be developed according to the strategies of the company and according to some management model, e.g. Balanced Scorecard. A data warehouse could also be used for implementation of

J. Grabis and M. Kirikova (Eds.): BIR 2011, LNBIP 90, pp. 44–58, 2011.

a Performance Measurement System. The related work in this field is discussed in following sections.

The definition of performance indicators should be based on the strategy of the company and for that purpose an appropriate method should be used. The indicators can be defined on various levels of formality. Some proposals on how to specify formally performance indicators exist and are described in [3], [4].

The authors of [3] propose a formal language for a modelling of goals based on performance indicators. The goal satisfaction could be controlled and the evaluation of the organizational performance could be performed. The authors of [4] propose a formal language for the indicator definition by introducing the sorts of indicators, predicates and functions included in it. Relationships between indicators could also be defined.

In our work we propose a formal specification of indicators that could be used in the performance measurement framework, which covers five-step indicator lifecycle and is also proposed in this paper.

The rest of the paper is organized as follows. The 2nd section introduces concepts of the performance measurement. The 3rd section describes performance measurement systems. The 4th section describes the performance measurement framework with an indicator lifecycle. The 5th section defines a requirement pattern and explains how it could be used for a formal definition of indicators. The 6th section ends the paper with conclusions and a description of the future work.

2 Performance Measurement Concepts

The definition of appropriate performance measures should be performed in a systemic way, based on well known approaches. The real world experience shows that companies use wrong measures [1], many of which are incorrectly treated as key performance indicators (KPIs). There is a lack of understanding what is and what is not a KPI, how success factors are connected with KPIs and organization's strategy.

2.1 Key Concepts: From Strategy to Measures

Before starting a discussion about formalization and choosing of appropriate performance measures, notions regarding performance measures have to be introduced.

Critical success factors (CSFs) [1] are issues or aspects of organizational performance that determine ongoing health, vitality and well-being. Usually from 5 to 8 CSFs are included in such list.

Success factors (SF) [1] are approximately 30 issues or aspects of organizational performance that are important in order to perform well in any given sector/industry. The most important of them are CSFs.

Performance measures [1] refer to indicators used by management to measure, report and improve the performance in an organization. Performance measures are classified as key result indicators, result indicators, performance indicators, or key performance indicators.

- Key result indicators (KRIs) represent summaries about many activities in an organization's CSF, but they do not help to understand what should be improved within organizations. KRIs can be financial and non financial.
- Result indicators (RIs) summarize some activities within CSF/SF, they are usually a result of more than one activity, but like KRIs they do not give information on what and how to improve. All financial performance measures are RIs,
- Performance indicators (PIs), on the contrary, „tell you what to improve" [1], because PIs measure a discrete activity. PIs are non financial.
- Key Performance Indicators (KPIs) „tell you what to do to increase performance dramatically" [1], KPIs represent the set of measures focusing on those aspects of organizational performance that are the most critical for the current and future success of the organization.

The set of used performance measures is influenced by *management models* of organizations, e.g. Balanced Scorecard (BSC) [5], and *measurement perspectives* defined within these models. In [5] four measurement perspectives are defined: Financial, Customer, Internal Process, and Learning and Growth. In [1] two more perspectives are added to the above mentioned in BSC – Environment/Community and Employee Satisfaction.

2.2 Features of Indicators

Indicators can be characterized according to different measurement aspects [1], [6].

- *Perspectives*. This aspect was already mentioned above. It should be added that not all indicator types cover all perspectives, e.g. financial perspective is related to KRIs and RIs, but not to PIs.
- *Time*. Subtypes of time aspect could also be considered – measurement periods when values of indicators are assigned and reporting periods that define the amount of historical data that should be included into reports. Although, KRIs are typically measured monthly, the time period for reporting may include even longer periods, e.g. year. KPIs are measured more frequently, e.g. daily or weekly.
- *Responsibility aspect*. Persons responsible for different types of performance measures could be at different levels starting from the top management to an individual level, where, for instance, in case of PIs all required actions are known and could be performed.
- *Activities*. RIs cannot be tied to a discrete activity; PIs, on the contrary, are tied to a discrete activity.
- *Success Factors*. Performance measures influence one or many CFS or SF depending on types of performance measures, e.g. KPIs impact more than one CSF /SF.
- *Reporting aspect*. The results for different types of performance measures could be reported at different levels starting from top management to individual level. For example, in case of KPIs results are reported to top management, they help to understand the required actions, and then the responsibility can be assigned to the individual level. So, Responsibility aspect and Reporting aspect not always have the same meaning.

2.3 Measurement Concepts

We can look at the performance measurement and indicators not only from the side of management models, but also from the point of view of measurement theories to find out appropriate definitions.

Measurement is [7] "the process by which numbers or symbols are assigned to attributes of entities in the real world in such a way as to characterize the attributes by clearly defined rules."

There exist ontologies, methods [8], [9], [10] for determination of indicators in some industries, e.g. Software process measurement. In business process measurement, the measurement is considered in the context of its applications, e.g. the business process reengineering or the performance measurement. The measurement concepts are not usually discussed in the case of business process measurement.

Further some definitions concerning the measurement are given to explain how the indicators are gained according to the measurement theories mentioned above.

Before the measurement, the information needs should exist, which are necessary for decision making. In the measurement, the following hierarchical structure is considered: attributes → base measures → derived measures → indicators → information.

A measured attribute is a property of an entity that can be identified.

A base measure is a measure of one attribute. The measurement method, if performed, assigns a value for this base measure.

A derived measure is defined as a function of two or more base or other derived measures.

An indicator is a measure that ensures the evaluation for particular attributes and is gained by means of an analysis which is performed according to an analysis model. Analysis model is an algorithm that combines two or more base and/or derived measures with decision criteria. Indicators provide the basis for a decision making and supply analysts with the necessary *information*.

Decision criteria are numerical goals that are used to estimate if certain activities should be performed or if further investigation of the situation is necessary.

3 Performance Measurement Systems

Companies use measurement systems to evaluate their performance. Management models, e.g. the Balanced Scorecard, should be used to develop such system to choose appropriate measures and to define the measurement framework. According to a management model, companies can be measured corresponding to different perspectives, e.g., financial, customer, internal business processes and others. Different aspects (e.g. connection to success factors, reporting and responsibilities) of performance indicators could be modelled and documented.

The advantage of using a data warehouse for the implementation of a performance measurement system is the possibility to use existing infrastructure of the company's data warehouse. Data warehouses reflect traditionally customer and financial indicators of companies. Management theories, e.g. the Balanced Scorecard, are hardly ever used in the field of data warehousing. We can conclude that internal business processes, learning and growth and other (e.g. environment/community) perspectives are typically not covered. The authors of [11] state that the next step is to integrate the relevant perspective of internal business processes into a data warehouse. Some experience in this direction is described also in [12], [13] and will be later briefly explained in this paper.

3.1 Data Warehouse as a Solution

Data warehouses as a solution for storing and analysing business data are described in a number of papers [13], [6]. This section summarizes the specific features of data warehouses, if they are used for an implementation of performance measurement systems.

- *Different new data sources.* Workflow logs are integrated with other data sources. As a special case only workflow log files are studied [14]. Workflow Management Coalition has defined three types of data that are related to workflow systems [15]: data of application programs, workflow-specific data and internal data of workflow systems. The workflow-specific data determines the choice of particular execution path; data is calculated at the moment of process execution and often not stored in workflow logs. Internal data of workflow systems is the data about the execution of the workflow; it is stored in log files.
- *Specific data analysis approaches.* Different ways for the analysis of workflow logs are described in [6], but they are applicable also in the case of other types of process data warehouses.

The data analysis could have different data analysis goals, e.g. technical or business goals. The examples of technical goals are testing of workflow systems, evaluation of response time or workload of a system. The data analysis could be performed at different levels by different users, so, personalized needs of an individual user or the interests of the whole company could be the focus of the data analysis. Individual users could have also different roles during the analysis, e.g. process owners, process performers, managers, support staff, data analyst and others.

The data analysis could have different time periods – e.g. short term analysis of processes, which presumes monitoring of process execution at the time of the analysis, and long term analysis (or process control), which means the analysis after the process execution.

These approaches could be combined, e.g. process monitoring in the case of technical goals could be used to evaluate the number of active users and the current workload. The data analysis also could have different analysis perspectives depending on the performance measurement framework used, e.g. customer, financial etc.

- *Data warehouse model.* The existence of different above mentioned aspects may determine the data items to be included into the data warehouse model for performance measurement. Important aspects that should be evaluated [13] are: the key measurements to be implemented in a data warehouse, their priorities and the situation with the necessary data to calculate these measurements. Possibly, some new data elements should be collected in the future.

3.2 Overview of Data Warehouses for Performance Measurement

Descriptions of implementations of performance measurement systems by means of a data warehouse are given in several works. Many solutions for specific aspects that arise, if data warehouses are used for implementation of performance measurement systems, are introduced (e.g. dimensional models).

The Process Data Warehouse is defined [12] "as a data warehouse which stores histories of engineering processes and products for experience reuse, and provides situated process support".

The concept of a Performance Management System (PMS) is defined in [13] as a system which "stores and manages all performance relevant data centrally, including both financial and non-financial data", and also ensures system's approach to measurement and timely access to data. The method used to build a data warehouse for the PMS is given. Performance indicators are defined based on analysis of company's goals, processes and stakeholders. Information needs are elicited. The PMS contains values of measurements and supplementary information about company structure, business processes, goals and performance measures. Besides traditional data warehousing perspectives of performance measurement, the process perspective is also analysed to some extent (e.g. execution time).

In [11] the authors propose a Corporate Performance Measurement System (CPMS), where process performance data is integrated with institution's data warehouse. Log files of a workflow system are used as data sources. Also, a method used to build a data warehouse for the CPMS is given. Goals for business processes are derived from goals of the company. Questions about measurement of goals are used, relevant indicators and data sources are described. The model of CPMS is developed as a part of an existing data warehouse model of the company.

3.3 Concept of a Data Warehouse of Processes

A category of data warehouses for performance measurement can be distinguished, where the focus is the storage and analysis of business process execution patterns. So, the concept of a Data Warehouse of Processes is introduced, however, the interpretations of the concept could differ.

In the systems mentioned already in previous section workflow data is used as a data source. The performance management system [13] stores process execution data besides other data to ensure the systematic measurement of processes. In the corporate measurement system [11], the data about the execution of processes from log files of workflow systems are integrated into the relevant company's data warehouse.

Workflow data warehouse [14] represents the concept of Data Warehouse of Processes. The authors of Workflow Data Warehouse [14] argue why and when data warehouse can be an appropriate solution for storing and analysing log files of process execution. Existing analysis tools provide limited possibilities – typical measures are number of executed process instances per time period, average execution time. The authors [14] propose a general-purpose model that is meant for storing different types of related facts, e.g. an activity is executed in the context of particular definition of workflow as a part of particular branch. The proposed model also includes Behaviour dimension that defines typical patterns of workflow execution in the past and that allows to analyse the current workflow execution, according to these predefined patterns. An important aspect of the proposed solution is that the dimensional model includes additional fact table Process Data Fact that represents the business data, where each particular process has changed.

3.4 Summary about Process Measurement Systems

On one hand, it is not enough to analyse separately some particular aspect, e.g. the workflow log files, because it does not help to evaluate the true status of the business, if we do not know anything about the data changed by business processes. On the other hand, when only business specific data is analysed, we can get information about a business situation, but there is a lack of reasonable information on why the situation is such as is shown in the business data analysis. The investigation of the workflow data can help to find out the bottlenecks of the workflow execution and to improve them.

Often in the real world not all possible data sources are used for data analysis, e.g. different log files, because of additional complexity of the integration; this data may also be underestimated as a valuable information source. The limited choice of data sources determine that the business indicators could not be freely defined.

A data warehouse as a solution for a process measurement system is appropriate, when mostly the long term analysis is performed, when integration of all possible data sources is needed and one part of necessary data already exists within a data warehouse of a company. A data warehouse will also be an appropriate solution, if analysis is performed at the level of an organization.

4 Performance Measurement Framework with Indicator Lifecycle

The previous concepts from the section about performance measurement concerning the management part of the measurement could be treated together with concepts from measurement theories to explain the nature of indicators in a most comprehensive way.

We could observe a lifecycle of indicators, which consists of 5 steps – indicator definition, measurement, analysis, reaction and improvement. Indicator definition step describes mostly different features of indicators that help to understand why that measure is introduced. The measurement step represents the process, when indicators get the values. The analysis step represents the process, when indicators are used to make decisions. The reaction step represents the process, when the decisions that are made in the previous step are implemented. The improvement step supports the evaluation of indicator definitions and values of aspects.

According to the five-step lifecycle of indicators, each step could be represented by different aspects (Figure 1). These aspects have the same meaning as described in the previous sections, but are grouped according to the particular step. We introduced some additional aspects, e.g. *Level*. This aspect describes the Level of indicators, whether it is needed for *Organization* level (team level) or for an *Individual* person. Concerning the *Process* aspect, it should be mentioned that for the reaction purposes also the goal of measurement should be clarified.

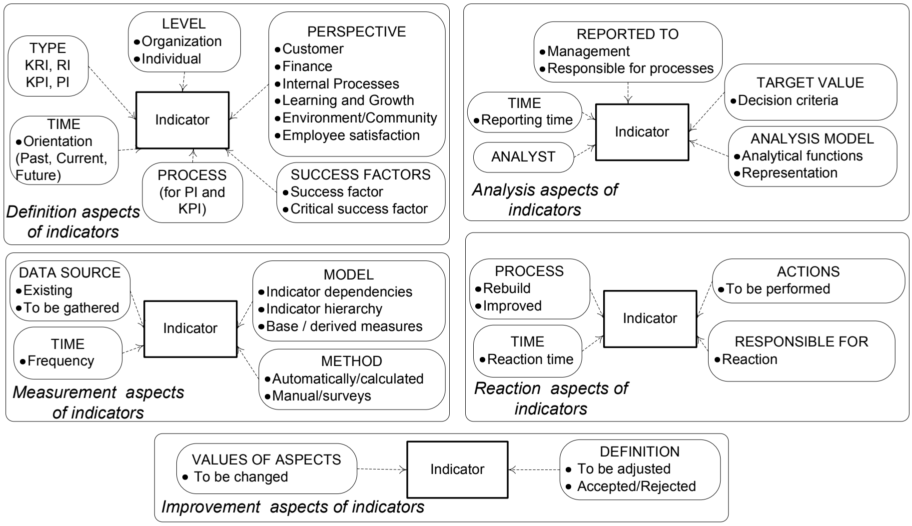


Fig. 1. Five groups of indicator aspects

The proposed measurement framework with indicators as the focus raises two further questions – (i) which is the most appropriate implementation to support the described lifecycle, and (ii) how the indicators should be formalized to bring the maximum of clarity into the process, what, why and how is measured.

The analysis of existing Performance Measurement Systems in Section 3 shows that the data warehouse can be chosen for a possible solution for the proposed measurement framework. Further in this paper, the formalization method of sentences that express the indicators is proposed.

5 Formal Model for Indicator Definition

Software requirements are normally described using a requirement specification language with various levels of formality [16]. The approach that we have proposed in our previous work [17] is based on extracting necessary information out of business requirements defined in natural language, thus, making business requirements more precise and formal. The proposed approach in [17] is meant for expressing requirements of data warehouse systems with one particular goal – to find out similar requirements for creating similar multidimensional model for a data warehouse that is built in the same industry.

The type of an information system to be developed has some impact on a way of formulating sentences that express requirements. Before starting our study, we assumed that requirements for data warehouses and information requirements particularly have a similar structure or pattern. This assumption was based on our observations on how the information needs were described in practical data warehousing projects. Also, the typical way the data warehouse data are analysed afterwards by means of typical OLAP operations with underlying typical SELECT statements shows that this assumption could be true.

Requirements are represented by sentences (subsets of words in natural language) that are used to specify what action(-s) the system should perform and which object (-s) it will affect. The question is whether the description of action(-s) and object(-s) could be a subject of formalization, e.g. how the terms of sentences are structured and if common pattern could be observed. In [17] a metamodel is given that describes the common structure of the information requirements for data warehouses.

In the case of performance measurement, we propose a revised metamodel defined in [17] to express the indicators.

We could use similar approach because of several reasons. On one hand, indicators are the focus of data analysis in the measurement process, according to the measurement concepts. On the other hand, the data warehousing models are built to represent the information needs for data analysis. So, we could talk about indicators as an information requirement for a data warehouse system. Therefore, the formalization of indicators could be based on the nature of elements of multidimensional models, e.g. the distinction between Quantifying Data and Qualifying Data.

We based the proposed model on the structure evaluation of the sentences that formulate performance indicators taken from the performance measures database [1].

5.1 Principles of Indicator Reformulation

After considering approximately 330 different indicators in [1] that refer to customer focus (CF), environment & community (EC), employee satisfaction (ES), finance (F), internal process (IP), and learning & growth (LG), we highlighted the following principles:

- An indicator component, which is supposed to be measured, is treated as an aggregated number of all occurrences of this component. For example, *calls* is reformulated to "count (call occurrence)", where *count* is the suitable aggregate function.
- If an indicator component is supposed to be shown in detail, then in the corresponding requirement the refinement function *show* is applied. For example, *employee* is reformulated to "show employee".
- If an indicator contains such components as "listing of", "list of", or "instances of", then in the corresponding requirement the refinement function *show* is applied. For example, *listing of customers* is reformulated to "show customers".
- If an indicator contains such component as "number of", then in the corresponding requirement the aggregate function *count* is applied. For example, *number of visits* is reformulated to "count (visit occurrence)".
- If an indicator contains such components as "cost of", "value of", "expense", "total expense", "income", "total income", "revenue", "investment", etc., or the name of currency in the beginning if the indicator, then in the corresponding requirement the aggregate function *sum* is applied. For example, *dollars saved* is reformulated to "sum (dollars)", however, *total income* is reformulated to "sum (income)".
- If an indicator contains such component as "average", then in the corresponding requirement the aggregate function *avg* is applied. For example, *average response time* is reformulated to "avg (response time)".

- If an indicator contains such components as "%", "percent", "percentage", or "ratio", then the percentage is substituted by *division* of partial quantity by total quantity. For example, an indicator *IT expense as a % of total administrative expense* is reformulated to "sum (IT expense) / sum (expense)".

Of course, the mentioned principles are supposed to be used taking into consideration the context of each indicator. One should analyse indicators to decide whether the data has to be aggregated or not and choose the appropriate aggregate function, if needed. Some of the instances of such indicators are: sales closed, initiatives completed, dates, candidates, days of production, energy consumed, etc.

5.2 Requirement Pattern Description

All indicators have common structure, for that reason it is possible to determine a pattern for re-writing business requirements formally. The approach that we use is describing indicators by means of formal grammar (EBNF notation) depicted in Figure 2. The same idea of requirement formalization may be represented as a metamodel. The metamodel is designed using UML 2.0 class diagram notation (Figure 3).

In Figure 2 business requirement is denoted by *Requirement* abstract class, which divides into a *Simple* and *Complex Requirement*. A complex requirement is composed of two or more simple requirements with an *Arithmetical Operator* between the simple requirements. A simple requirement consists of a verb (*Operation*) that denotes a command, which refers to an *Object*, and zero or one *Typified Condition*.

There are two kinds of data in data warehousing: *Quantifying* (measurements) and *Qualifying Data* (properties that characterize measurements). An object is either an instance of quantifying or qualifying data depending on the requirement.

The term "operation" describes the kind of *Action(-s)* to be performed. If some kind (or different kinds) of action should be performed more than once, then it is called a *Complex Operation*. We propose two possible types of action: an *Aggregation* (a command, used for calculation and grouping, "roll-up") and a *Refinement* (a command, used for information selection, "drill-down", as an opposite to an aggregation). Information refinement is either showing details, i.e., selecting information about one or more objects, or slicing, i.e., showing details, according to a certain constraint (*Typified Condition*).

If there is a restriction in the requirement, then it is represented by a typified condition. There are two types of conditions: *Simple Condition* and *Complex Condition*.

Complex condition joins two or more simple conditions by *Logical Operators* (such as "and", "or", "not"). Simple condition consists of a *Comparison* of two *Expressions*, for example, "time is greater than last_access_time – 1 second". An expression as well may be either a *Simple Expression* or a *Complex Expression*. A complex expression contains two or more simple expressions with an arithmetical operator between the simple expressions. A simple expression belongs either to qualifying data (for example, "last_access_time") or to *Constants* (for example, "1 second").

<Requirement>: <Simple Requirement> | <Complex Requirement>
<Complex Requirement>: <Requirement> <Arithmetical Operator> <Requirement>
<Arithmetical Operator>: + | - | * | /
<Simple Requirement>: <Operation> <Object> <Typified Condition>?
<Object>: <Quantifying Data> | <Qualifying Data>
<Operation>: <Complex Operation> | <Action>
<Complex Operation>: <Operation>+
<Action>: <Aggregation> | <Refinement>
<Aggregation>: count | sum | average | …
<Refinement>: show
<Typified Condition>: <Condition Type> <Condition>
<Condition Type>: where
<Condition>: <Simple Condition> | <Complex Condition>
<Complex Condition>: <Condition> <Logical Operator> <Condition>
<Logical Operator>: or | and | not
<Simple Condition> : <Expression> <Comparison> <Expression>
<Comparison> : > | >= | < | <= | =
<Expression> : <Simple Expression> | <Complex Expression>
<Complex Expression>: <Expression> < Arithmetical Operator> <Expression>
<Simple Expression>: <Qualifying Data> | <Constant>

Fig. 2. Business requirements in EBNF notation

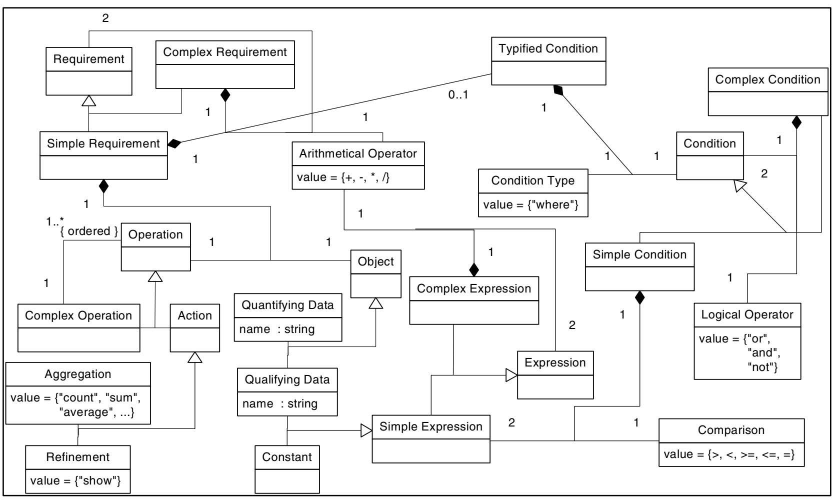


Fig. 3. Formalized requirements metamodel (UML)

5.3 Indicator Examples

Examples given in this section illustrate the application of proposed formal model to define the indicators. We use performance indicator definitions formulated in natural language from a performance measures database [1]. This database contains a comprehensive list of performance measures from six measurement perspectives, e.g. customer, finance, etc. We have chosen indicators from different perspectives with different structure of the sentences in a natural language.

<table>
<tr><td>show</td><td>Refinement</td><td>Action</td><td colspan="2">Operation</td><td rowspan="8">Simple Requirement</td></tr>
<tr><td>month</td><td>Qualifiyng Data</td><td colspan="3">Object</td></tr>
<tr><td>AVG</td><td>Aggregation</td><td>Action</td><td colspan="2">Operation</td></tr>
<tr><td>contacts ocurrance</td><td>Quantifying Data</td><td colspan="3">Object</td></tr>
<tr><td>where</td><td colspan="3">Condition Type</td><td rowspan="4">Typified Condition</td></tr>
<tr><td>customer type</td><td>Qualifiyng Data</td><td>Simple Expression</td><td rowspan="3">Simple Condition</td></tr>
<tr><td>=</td><td colspan="2">Comparison</td></tr>
<tr><td>'key customer'</td><td>Constant</td><td>Simple Expression</td></tr>
</table>

Fig. 4. Requirement formalization example of the requirement type CF

Suppose that we would like to obtain information about average number of contacts made with key customers per month. This indicator represents the indicators regarding customer focus. This statement could be reformulated using our proposed requirement pattern metamodel (Figure 3). A statement that is “valid”, i.e., can be derived from the original indicator, is “show month, average (contact occurrence) where customer type is ‘key customer’ ”. Figure 4 demonstrates the application of requirement patterns taking as an example the business requirements mentioned above. The left column is filled with parts of the statement and all the rest columns (left to right) contain names of the metamodel levels from the bottom to the top.

Let’s take other examples from the list of indicators. Suppose that we would like to obtain information about late deliveries to key customers. This indicator represents the indicators regarding customer focus and internal processes. A statement that is derivable from the metamodel (Figure 3) is “show deliveries where delivery type is ‘late’ ”. Because of the space limitations, Figure 5.a demonstrates only the bottom level elements of our proposed requirement pattern metamodel (2nd column) and corresponding parts of the statement (1st column).

Consider that we would also like to get information about staff turnover by type (resignations, end of contract, temporary staff, and termination). This indicator represents the indicators regarding employee satisfaction. A statement that can be derived, using our proposed requirement pattern metamodel (Figure 3), is “show employment types, count (person occurrence) where employment type is ‘resignation’ or ‘end of contract’ or ‘temporary staff’ or ‘termination’ ”. Figure 5.b demonstrates the bottom level elements of our proposed requirement pattern metamodel and corresponding parts of the statement.

Suppose that we would like to obtain information about the number of sponsorship projects in past 12 months by company. This indicator represents the indicators regarding the environment and community focus. Let’s assume that the time period closes with current date. A statement that can be derived, using our proposed requirement pattern metamodel (Figure 3), is “show company, count (project occurrence) where project type is ‘sponsorship’ and date is greater than current_date - 356 and date is less than current_date”. This requirement involves also an arithmetical operator, which is used in the expression together with qualifying data and a constant. Therefore, the formalization example is more complicated than the previous examples. Figure 6.a demonstrates the bottom level elements of our proposed requirement pattern metamodel and corresponding parts of the statement.

(a)

	Indicator Component	Metamodel Element
1	show	Refinement
2	delivery	Qualifying Data
3	where	Condition Type
4	delivery type	Qualifying Data
5	=	Comparison
6	'late'	Constant

(b)

	Indicator Component	Metamodel Element
1	show	Refinement
2	employment type	Qualifying Data
3	count	Aggregation
4	person occurrence	Quantifying Data
5	where	Condition Type
6	employment type	Qualifying Data
7	=	Comparison
8	'resignation'	Constant
9	or	Logical Operator
10	employment type	Qualifying Data
11	=	Comparison
12	'end of contract'	Constant
rows 9-12 are repeated for 2 more comparisons with different constants		

Fig. 5. Requirement formalization examples: (a) Type – CF, IP; (b) Type – ES

(a)

	Indicator Component	Metamodel Element
1	show	Refinement
2	company	Qualifying Data
3	count	Aggregation
4	project occurrences	Quantifying Data
5	where	Condition Type
6	project type	Qualifying Data
7	=	Comparison
8	'sponsorship'	Constant
9	and	Logical Operator
10	date	Qualifying Data
11	>	Comparison
12	current_date	Qualifying Data
13	-	Arithmetical Operator
14	356	Constant

(b)

	Indicator Component	Metamodel Element
1	sum	Aggregation
2	expense	Quantifying Data
3	where	Condition Type
4	expense type	Qualifying Data
5	=	Comparison
6	'IT'	Constant
7	/	Arithmetical Operator
8	sum	Aggregation
9	expense	Quantifying Data

Fig. 6. Requirement formalization examples: (a) Type – EC; (b) Type – F

Assume that we are interested in summary information on the percentage of IT expense of total administrative expense by quarters in a year. This indicator represents the indicators regarding finance. A statement that can be derived, using our proposed requirement pattern metamodel (Figure 3), is "(sum (expense) where expense type is equal to 'IT') divide by (sum (expense))". This requirement is complex, and it is composed of two simple requirements and an arithmetical operator. Here we apply arithmetical operator '/' to calculate percentage by dividing expenses in IT by total expenses. Figure 6.b demonstrates the bottom level elements of our proposed requirement pattern metamodel and corresponding parts of the statement.

6 Conclusions and Future Work

We have already executed performance measurement tasks at our University, and we have described the approaches and techniques used for that purpose in our previous works [18], [19]. A process measurement and monitoring system (PMMS) was created; a data warehouse was used as one of the important elements of this solution. The PMMS consists of: 1) a process operational monitoring component, 2) a process measurement system, and 3) a process execution log file. The process operational monitoring component supports the analysis of indicators of the process workflow directly from the log file during the process execution. Then the indicator's data is loaded into a data warehouse and used for the process measurement. The results are provided in the case studies of our approach in [18].

Despite the fact that the PMMS is successfully used, we have searched for a more systemic approach to facilitate a more targeted measurement. As a result, the measurement framework proposed in this paper is described, and it is obvious that we will complement the existing PMMS, according to the proposed framework. Also, a data warehouse will serve as an implementation platform for the new version of the PMMS. Not only our experience, but also our literature studies showed that a data warehouse that already exists in an organization could be effectively used for performance measurement purposes.

Another important part of our research is a model for indicator formalization. The future work will be done in two directions. The first one is concerning practical case studies and evaluations of ease of use of formal patterns. The second direction is a development of a new method for semi-automated construction of data warehouse schemas based on the formal definitions of indicators, according to the model given in this paper.

Acknowledgments. This work has been supported by ESF project No. 2009/0216/1DP/1.1.1.2.0/09/APIA/VIAA/044.

References

1. Parmenter, D.: Key Performance Indicators: Developing, Implementing, and Using Winning KPIs, 2nd edn. John Wiley & Sons, Inc., Chichester (2010)
2. Harrington, J.H.: Business Process Improvement – The Breakthrough Strategy for Total Quality, Productivity, and Competitiveness. McGraw-Hill, New York (1991)
3. Frank, U., Heise, D., Kattenstroth, H., Schauer, H.: Designing and Utilizing Business Indicator Systems within Enterprise Models - Outline of a Method. In: Loos, P., Nüttgens, M., Turowski, K., Werth, D. (eds.) Modellierung Betrieblicher Informationssysteme (MobIS 2008) - Modellierung zwischen SOA und Compliance Management. LNI, vol. 141, pp. 89–105. Saarbrücken, Germany (2008)
4. Popova, V., Treur, J.: A Specification Language for Organizational Performance Indicators. Applied Intelligence Journal 27(3), 291–301 (2007)
5. Kaplan, R.S., Norton, D.P.: The Balanced Scorecard. Harvard Business School Press, Boston (1996)
6. Muehlen, M.: Process-driven Management Informations Systems – Combining Data Warehouses and Workflow Technology. In: ICECR-4 (2001)

7. Fenton, N.E., Whitty, R.: Software Quality Assurance and Measurement: A Worldwide Perspective, pp. 1–19. International Thomson Computer Press (1995)
8. Munson, J.: Software Engineering Measurement. Auerbach Publications (2003)
9. McGarry, J., Card, D., Jones, C., et al.: Practical Software Measurement: Objective Information for Decision Makers. Addison-Wesley, Reading (2001)
10. Garcıa, F., Bertoa, M., Calero, C., et al.: Towards a Consistent Terminology for Software Measurement. Information and Software Technology 48(8), 631–644 (2006)
11. List, B., Machaczek, K.: Towards a Corporate Performance Measurement System. In: Proceedings of the ACM Symposium SAC 2004, pp. 1344–1350. ACM Press, New York (2004)
12. Jarke, M., List, T., Koller, J.: The Challenge of Process Data Warehousing. In: Proceedings of the 26th International Conference VLDB 2000, pp. 473–483 (2000)
13. Kueng, P., Wettstein, T., List, B.: A Holistic Process Performance Analysis through a Process Data Warehouse. In: Proceedings of the 7th American Conference on Information Systems (AMCIS 2001), pp. 349–356 (2001)
14. Bonifati, A., Casati, F., Dayal, U., Shan, M.C.: Warehousing Workflow Data: Challenges and Opportunities. In: Proceedings of the 27th International Conference VLDB 2001, pp. 649–652 (2001)
15. Workflow Management Coalition,Terminology and Glossary, 3rd edn., Document Nr. WFMC-TC-1011, Winchester (1999)
16. Videira, C., Silva, A.R.: Patterns and Metamodel for a Natural-language-based Requirements Specification Language. In: Pastor, Ó., Falcão e Cunha, J. (eds.) CAiSE 2005. LNCS, vol. 3520, pp. 189–194. Springer, Heidelberg (2005)
17. Kozmina, N., Niedrite, L., Solodovnikova, D.: A Knowledge-based Method for Transforming Requirements to Conceptual Model of a Data Warehouse. In: Proceedings of the 8th International Baltic Conference on Databases and Informational Systems (DBIS 2008), Estonia, pp. 63–74 (2008)
18. Medvedis, I., Niedrite, L., Niedritis, A., et al.: e-University Initiative: Approach, Solutions and Case Studies. In: Proceedings of the 8th International Baltic Conference on Databases and Information Systems, Estonia, pp. 141–152 (2008)
19. Niedrite, L., Solodovnikova, D., Treimanis, M., Niedritis, A.: The Development Method for Process-Oriented Data Warehouse. In: Mastorakis, N.E. (ed.) WSEAS Transactions on Computer Research, vol. 2(2), pp. 183–190 (2007)

Using the Entity-Attribute-Value Model for OLAP Cube Construction

Peter Thanisch[1], Tapio Niemi[2], Marko Niinimaki[2], and Jyrki Nummenmaa[1]

[1] Department of Computer Sciences, FIN 33014 University of Tampere, Finland
{pt,jyrki}@cs.uta.fi
[2] Helsinki Institute of Physics, Technology Programme, CERN,
CH-1211 Geneva 23, Switzerland
{tapio.niemi,man}@cern.ch

Abstract. When utilising multidimensional OLAP (On-Line Analytic Processing) analysis models in Business Intelligence analysis, it is common that the users need to add new, unanticipated dimensions to the OLAP cube. In a conventional implementation, this would imply frequent re-designs of the cube's dimensions. We present an alternative method for the addition of new dimensions. Interestingly, the same design method can also be used to import EAV (Entity-Attribute-Value) tables into a cube. EAV tables have earlier been used to represent extremely sparse data in applications such as biomedical databases. Though space-efficient, EAV-representation can be awkward to query.

Our EAV-to-OLAP cube methodology has an advantage of managing many-to-many relationships in a natural manner. Simple theoretical analysis shows that the methodology is efficient in space consumption. We demonstrate the efficiency of our approach in terms of the speed of OLAP cube re-processing when importing EAV-style data, comparing the performance of our cube design method with the performance of the conventional cube design.

Keywords: OLAP, dimensions, EAV.

1 Introduction

Business Intelligence users utilise data from different sources for their data analysis needs. A classical approach is based on analysing the analysis needs to build processes to load data into a data warehouse and then to construct the analysis data from the data warehouse. The analysis data may be structured as a simple table or a pivot table, or, for more complicated analysis, as a multidimensional OLAP (On-Line Analytic Processing) cube.

While it is not so common for the operational information systems to be in constant change, new analysis needs come in frequently, due to e.g. new changes in the environment in which the business takes place, or just simply due to new analysis ideas. These new analysis needs may well involve the use of unanticipated data.

J. Grabis and M. Kirikova (Eds.): BIR 2011, LNBIP 90, pp. 59–72, 2011.

Technically, this can be done by combining data from external data sources, but it is a non-trivial task to merge the new data with the existing design. Updating the dimensional design and re-processing the cube would be sufficiently resource-consuming so as to preclude train-of-thought working. We explore an alternative approach that allows an element of data-driven dynamical approach to dimensional design.

Our method is based on the idea of integrating the new data into the design model by using EAV (Entity-Attribute-Value) tables. The classic use of EAV tables has been to avoid extremely sparse tables in application areas such as biomedical analysis. Sparsity, in general, is problematic in OLAP design, since it can heavily increase the amount of aggregation computation and storage space for aggregation results [7]. We will show how the EAV tables can be used to integrate new data in a natural way into the multidimensional design and in this way also help with the sparsity problem. We point out that our technique for importing EAV data only works for categorical/descriptive data and not for fact values.

To demonstrate the value of handling EAV data, we show that we can work directly from an EAV table, without having to expand it out to a conventional table. Furthermore, we do not need to create any special indexing structure.

Our paper is organised as follows. In Section 2 we review related work. Section 3 demonstrates our methodology of converting data into a format that supports our model, while Section 4 describes an example implementation. Section 5 contains a comparison of cube construction performance between our solution and a conventional approach. Finally, Section 6 contains the final conclusions.

2 Related Work

In the Entity-Attribute-Value (EAV) data model [6,2], a row in a table stores a single fact describing an entity. The fact comprises (1) a key value identifying the entity, (2) an attribute (typically a foreign key to a reference table containing descriptive data and metadata about the attribute) and (3) the value of the attribute. The motivation behind the EAV data model is (a) to handle sparse data efficiently and (b) to simplify the database schema in applications where, for each individual entity, a large number of the attributes are not applicable to that entity.

EAV's similarities with the well-known Resource Description Framework (RDF) [5] are obvious: both of them can present data in terms of triples. We assume that the EAV/CR (Entity-Attribute-Value with Classes and Relationships) and RDF are equivalent, though to our knowledge there is no formal proof of it. There are also similarities between EAV and the associative model of data [9].

Russo [8] implements a "Multiple Groups" model to handle the case where a single dimension member can belong to more than one group and where it is not possible to define, in advance, the set of groups to which the dimension member belongs before the analysis starts. Russo's Multiple Groups model is based on a fixed relational and multidimensional schema. The model handles the case in which newly-loaded data defines new groups that are immediately available in the analysis. Furthermore, these newly-defined groups can be used in the definition of yet more groups.

Leonhardi et al. [4] address the problem that, in a conventional OLAP system, the set of dimensions used to analyse the multi-dimensional data set is fixed. Leonhardi et al. have implemented an OLAP environment that provides facilities for the dynamic and interactive creation of dimensions, based on multi-dimensional grouping mechanisms. This allows the user to combine several dimensions. Furthermore, the newly-defined groups can themselves be refined to create yet more groups.

Finally, Lee and Shim [3] compare storing information using two schemata: a horizontal (or conventional) schema and a "vertical schema" that seems natural for storing Entity-Attribute-Value data. They present pivoted table indices to improve the query performance of the vertical schema. However, they have not measured the OLAP cube construction performance of the schemata.

3 Methodology

In this work, we provide a fairly general-purpose model, which we call 'dimensionless OLAP', whereby EAV data can be loaded into an OLAP model with minimal intermediate processing. The same model can also be used to implement many-to-many relationships. The idea is based on embedding multiple groups in a single dimension table: it creates a single bridge and a single outer dimension table. The bridge table contains the information (in the form of a foreign key relationship) which identifies the group or groups associated with each of dimension members.

We are interested in the scenario in which the member information for one of the dimensions in a multidimensional database is stored as an EAV table in the source relational database system. We assume that the dimension has a very large number of attributes and that a conventional representation of the relation in the source system would be extremely sparse.

In the context of our current work, we can interpret the "E" in EAV in two ways:

1. "E" stands for "entity". This case represents for example the biomedical EAV data. The 'A's, i.e. attributes, are just names of columns in the tabular representation of the entity.
2. "E" stands for "entity class". This is Russo's "multiple groups" presentation. Again, the 'A's are names of columns, but in this time, columns provide grouping categories in a hierarchy.

The difference in the two interpretations of EAV is the way in which many-to-many relationships are handled. In the E-stands-for-entity interpretation, there is a many to one relationship between Attribute and Value. However, for the E-stands-for-entity-class interpretation, the relationship can be many-to-many.

The benefit of our method is that we still use many-to-many relationships, even with the EAV-style data. This is not possible in a conventional relational OLAP system without some additional facility for modelling many-to-many relationships or handling this sort of relationship. Another benefit is that we do not need to expand the EAV data out into a conventional tabular structure and we do not need to pivot the data to build a special index structure as, for example, Lee and Shim did [3].

Our model is based on the following tables:

– **Fact Table.** The fact table holds the quantitative information that forms the subject matter of the multidimensional database. In addition, each row in the fact table references the corresponding members in the other dimensions by way of columns holding foreign keys, with one foreign key column for each dimension.
– **Inner Dimension Table.** The inner dimension table has one row for each member of the dimension. It contains other attributes describing the member. The identity of a member corresponds to the entity values in the EAV model.
– **Bridge Table.** The bridge table is the mechanism for implementing many-to-many relationships. It contains two columns, one which references a member in the inner dimension tables and the other referencing a record in the outer dimension table.
– **Outer Dimension Table.** The outer dimension table can contain several rows for each dimension member. One of the columns identifies the entity so that the bridge table can reference the correct row. In conventional multidimensional databases, the other columns simply contain attribute information for the entity instance in the many-to-many relationship with the dimension member. However, in our implementation, the outer dimension table contains the EAV data and there are only two other columns in the outer dimension table, one containing the name of an attribute and the other containing the value of the attribute for this particular dimension member.

When discussing the sizes of the tables, we ignore data types and make the simplifying assumption that the size of a table can be approximated by its number of cells (i.e. row/column intersections). Suppose that the original EAV table has n rows and that the E column has e distinct entities and the A column has a distinct attributes. Then expanding out the EAV table to a conventional representation would result in a table of size $e \times a$ cells. The size of the EAV table is $3 \times n$ cells, where n is the number of entity/attribute combinations with non-Null values. So long as the conventional table is sufficiently sparse (i.e. there is a sufficiently large number of cells with a NULL, rather than a value) then $3 \times n < e \times a$.

We must establish that the storage requirements for our approach is a of the order of $3 \times n$. First, we note that the fact table contains information that is separate from the EAV table information. The EAV table simply holds the members of a dimension and their associated attribute information.

The inner dimension table contains e rows. Clearly, e ≤ n, since the EAV table has at least one row per dimension member.

The outer dimension is simply the EAV table itself, hence it has n rows. (In our implementation, we have added a surrogate key column, but this is only for convenience.) We note that the size of the outer dimension could be considerably reduced, since all that is required in the outer dimension is the set of distinct attribute/value pairs for the dimension. Only the bridge table needs to contain the entity information.

The bridge table contains n rows, one row for each distinct entity/attribute pair in the EAV table. Hence the tables in the multidimensional design have no more rows than the EAV table and each of the tables has a fixed number of columns.

4 Example Implementation

We designed a test database, which is like the Russo's "Multiple Groups" database, but much larger and with more outer dimension tables and bridge tables. There are 10 outer dimension tables (each representing a group) and 10 bridges in the database, in addition to the fact table and other dimension tables. We populated all of the tables with synthetic data as illustrated in Table 1.

Table 1. Source EAV table

SK	ENT	ATT	VAL
1	E00001	A00001	V001
2	E00001	A00002	V002
3	E00001	A00002	V003
4	E00002	A00003	V004
:	:	:	:
20	E00010	A00011	V020

In this example, the table contains information about ten different entities (E00001 – E00010). This information is spread over eleven different attributes. However, in our example table, each entity has non-NULL values for just two of those attributes. We pose the question: is it possible to transfer this information into the multidimensional database without having to unpack the EAV table into the conventional representation?

Using the facilities in Microsoft SQL Server 2008 [1], we have the following simplified target architecture with four tables as shown in Figure 1. Since in our simplified diagram we have only one cube dimension, the fact table has only one foreign key to dimension tables. The inner dimension table has one row for each member of the dimension. In our simplified example, it comprises a single column which contains the identity of each member in the dimension. The bridge table contains two columns, one which references a member in the inner dimension tables and the other referencing a record in the outer dimension. The outer

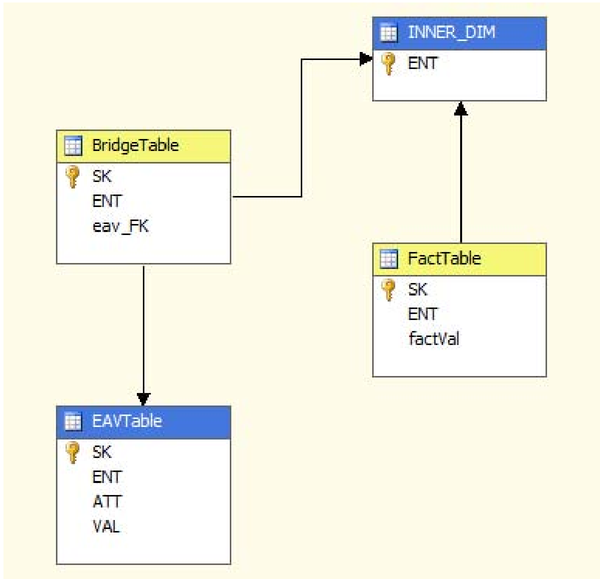


Fig. 1. Database diagram (The arrows correspond to foreign key relationships between the corresponding tables.)

dimension table is named as EAVTable in our example. It contains a reference to the bridge table and there are only two other columns in the outer dimension table, one containing the name of an attribute and the other containing the value of the attribute for this particular dimension member.

With regard to the cube design, the procedure that we followed is consistent with the minimum effort needed to obtain useable output. We note that the following description might appear to be convoluted. This is because we are treating the entity, attribute and value fields from the EAV table as attributes in our dimensional model. However, as we shall see, we must assign these three EAV fields different roles in the dimensional model in order for the model to work correctly.

For the dimension design of EAVTable, we defined the attribute relationships as depicted in Fig. 2.

This allowed us to define a user hierarchy for the dimension; see Fig. 3.

In this special implementation, we want the value from the EAV table to be associated with the attribute, rather than being treated as a separate dimension attribute in its own right. Fortunately, there is a facility to do just this in the dimension designer. We defined both ENT and ATT to be the key for ATT and we defined VAL to be the value field associated with ATT.

With regards to the cube design, fortunately SQL Server's cube designer is able to deduce most of what is needed automatically from its analysis of the foreign key relationships between the tables in the underlying relational database.

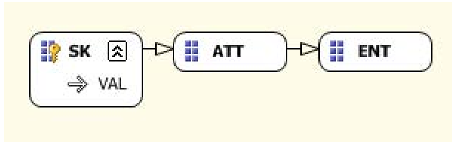


Fig. 2. Attribute relationship

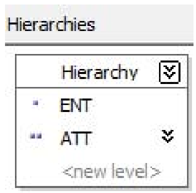


Fig. 3. User hierarchy

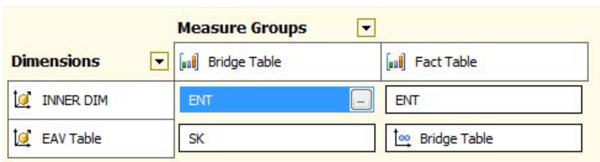
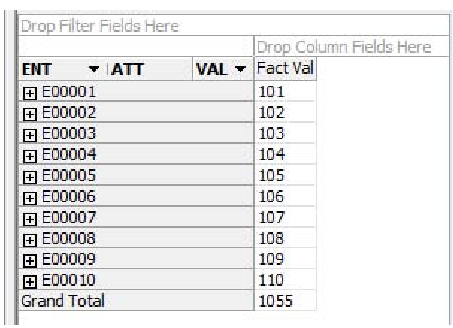


Fig. 4. Cube Designer: Specifying the many-to-many relationships



Drop Filter Fields Here

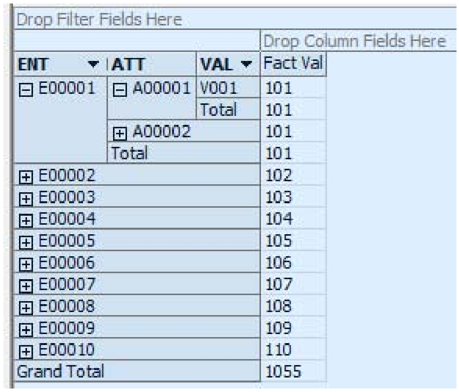
Drop Column Fields Here

ENT	ATT	VAL	Fact Val
E00001			101
E00002			102
E00003			103
E00004			104
E00005			105
E00006			106
E00007			107
E00008			108
E00009			109
E00010			110
Grand Total			1055

Fig. 5. Browsing the cube

We specified that INNER_DIM and EAVTable are to be used as dimension tables and that FactTable and BridgeTable are to be used as fact tables. (In traditional terminology, a bridge table is a factless fact table.)

Foreign key relationships are defined from FactTable to INNER_DIM, from BridgeTable to INNER_DIM and from BridgeTable to EAVTable. SQL Server's



Drop Filter Fields Here

Drop Column Fields Here

ENT	ATT	VAL	Fact Val
⊟ E00001	⊟ A00001	V001	101
		Total	101
	⊞ A00002		101
	Total		101
⊞ E00002			102
⊞ E00003			103
⊞ E00004			104
⊞ E00005			105
⊞ E00006			106
⊞ E00007			107
⊞ E00008			108
⊞ E00009			109
⊞ E00010			110
Grand Total			1055

Fig. 6. Cube browsing: expanded results tree

cube designer shows that the relationships depicted in Fig. 4 exist between the dimension tables and the fact tables, which it refers to as measure groups. In particular, we observe that the cube designer detected automatically that there is a many-to-many relationship between EAVTable and FactTable.

Browsing the cube, we get the results shown in Figure 5. As in our simple example we have just one dimension, there is one row for each of the ten entities in the EAV table. We can examine the attribute values by clicking to expand the tree, as the diagram in Figure 6 illustrates.

5 Tests and Results

We compare a conventional OLAP design with a "multiple groups" design in terms of performance, where performance is defined in terms of cube processing time. The performance comparison was between the following implementations:

1. For the classic approach, when a new set of groups needs to be added to the analysis, this means adding a bridge table and a dimension table. The cube must be partially re-processed.
2. For the multiple groups approach, new rows are added to the bridge table and the outer dimension table. The only cube processing necessary is incremental processing for the newly-added rows.

We looked at cube processing time for the "classic" implementation of multiple groups and compared it to cube processing time for our approach. Our tests showed that cube processing gets progressively worse, relatively and absolutely, as the number of dimensions increases in the conventional design. The key difference is that the classic approach involves adding an extra dimension and an extra bridge to the design, whereas our approach just involves adding rows to an existing dimension and bridge table.

In this performance measurement exercise, we used a 64-bit system running under Microsoft Windows 7 Professional. The processor was an Intel(R) Core(TM) i3 CPU M330 @ 2.13GHz (dual core) processor and there was 4GB of RAM. The database management system used was Microsoft SQL Server 2008 R2 (64-bit). In SQL Server, the OLAP cubes were stored as MOLAP (multi-dimensional OLAP) structures.

The performance test scenario, which is based on Russo's Multiple Groups example [8], comprises two alternative ways to implement a facility whereby a user can provide supplementary descriptive categorical data for the existing members of an existing cube dimension. In particular, it is assumed that these newly-added data items can be in a many-to-many relationship with the members in the existing dimension, hence the scenario cannot be implemented by using a simple reference (or outer dimension table) table or by adding some columns to the existing dimension table. Instead, it is necessary to use one or more bridge tables to implement the many-to-many relationship. We call the two different approaches to implementing this facility the multiple bridges design and consolidated bridge design. To simplify the scenario, and in particular the performance comparison, we assume that all of this activity is focussed on just one of the dimensions.

In this scenario, it is assumed that the user looks at some preliminary results from the cube and then decides to add some more descriptive data. For the purposes of the scenario, this user behaviour is assumed to occur repeatedly. One consequence of the way that we have defined the scenario is that the key performance element is likely to be cube processing, rather than query processing. For large cubes, cube processing can take hours, disrupting the train of thought of the user and increasing the delay in the user producing the results. In such applications, it may be that only a few queries are posed against the cube whilst it is in a particular state before the user determines that yet another change is necessary. Hence we focus on cube processing time as the key metric in our performance comparison.

In brief, the multi-bridge technique is the obvious implementation: each time that the user wants to add more supplementary data to a dimension, a new bridge table is added to the cube design along with the new dimension table containing the supplementary data. Each such bridge table acts as the intermediary between the new dimension table and the existing dimension.

In the consolidated bridge implementation, no new tables are added to the design. Instead, a single new dimension table and a single new bridge table handle all of successive increments of information. In this implementation, instead of adding a new bridge table and a new dimension table, the information is inserted into the existing bridge and dimension tables. However, all of the information that would otherwise have been added in the new dimension must now be encoded on the existing bridge table and outer dimension table. Interestingly, this information must include the name of the relation. In the conventional implementation, the name of the relation would show up as the name of a dimension in the cube design. In our design, we have to include the name of the relation in the attribute data so that it can be used for filtering, etc. in browsing and reporting applications that use the cube.

Both implementations use a single inner dimension, which we have named DIM_TBL, and a fact table, FACT_TBL. These two tables remain unchanged in both implementations of this scenario. Both tables have 10,000 rows in this performance comparison. As usual, each row in the fact table has a column containing a foreign key reference to dimension member in DIM_TBL.

5.1 Multiple Bridges

In the multiple bridges approach, each new dimension table has just two columns, a surrogate key and the text column containing the descriptive information. Each such table represents one of the groups and we have named the tables GRP_001, GRP_002, etc. We refer to these tables as the outer dimension tables. The column containing the text value is named GRP_VAL.

We named the bridge tables for this approach as BRI_001, BRI_002, etc. Each bridge table contains a column FK_GRP which contains a foreign key reference to a row in the corresponding GRP_ table and a column FK_GENDIM which contains a foreign key reference to a row in DIM_TBL.

For the purposes of our performance comparison, each outer dimension table has 100 rows and each bridge table has two thousand rows. The diagram in Figure 7 shows the multiple bridges design where there are five pairs of bridges and outer dimension tables.

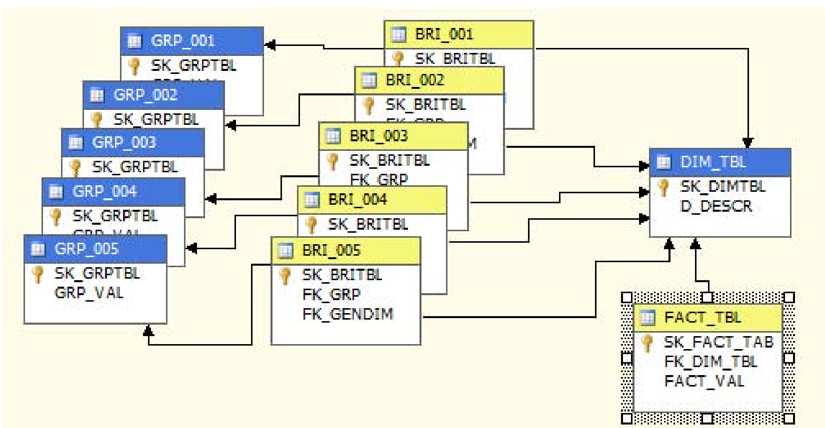


Fig. 7. Multiple bridges design

In the performance test, the number of bridge/outer dimension pairs was incremented from four to ten. At each increment, the cube was re-processed and the cube processing time (both CPU time and duration) was measured using the profiling tool provided with SQL Server.

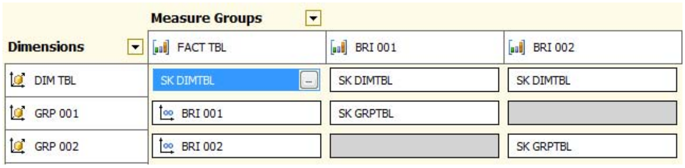


Fig. 8. Adding new bridge tables

For the user, adding a new dimension table and bridge table is very simple. The many-to-many relationship between the fact table and the outer dimension table must be specified using the dimension designer, but this was also straightforward. The dimension design where there are just two outer dimension/bridge pairs as shown in Figure 8.

5.2 Consolidated Bridge

In the multiple bridges approach there is just one bridge table/outer dimension table pair. When the user needs to add supplementary information, it gets inserted into this existing pair of tables (instead of changing the design by adding a new outer dimension/bridge pair, as in the multiple bridges implementation).

In this implementation, we named the bridge table as BRIDGE_TBL and the outer dimension table as EAV_TBL. As well as a surrogate key column, EAV_TBL has three columns, GEN_ENT, GEN_ATT and GEN_VAL. We should stress that this table is not identical in information content and structure to tables in the EAV model. In our implementation, the GEN_ENT column, each row actually contains the name of an entity class, rather than an entity. Each distinct entity class corresponds to one of the outer dimension tables in the Multiple Bridges implementation. The GEN_VAL column contains the values that occur in the GRP_VAL columns of the outer dimension tables in the Multiple Bridges implementation.

Unlike the EAV model, we do not need an attribute column, since the information we model comes from tables that have only got one information column.

The point about this implementation is that the set of entities with which the information in EAV_TBL is associated is the set of entities in the inner dimension table. However, the association with these entities is not explicit in EAV_TBL itself. Rather, the association is provided by BRIDGE_TBL.

The following query retrieves the rows in the bridge table associated with a particular member in the inner dimension.

SELECT * FROM BRIDGE_TBL WHERE FK_GENDIM = 1
The results are shown in Figure 9.

Now we retrieve the rows from EAV_TBL that are referenced by the bridge table:
SELECT * FROM EAV_TBL WHERE SK_EAV = 46 OR SK_EAV = 92
The results are as follows shown in Figure 10.

	PK_BRIDGE_TBL	FK_EAV_TBL	FK_GENDIM
1	1	46	1
2	2	92	1

Fig. 9. Results of bridge table query

	SK_EAV	GEN_ENT	GEN_VAL
1	46	GRP_001	GRP_001-46
2	92	GRP_001	GRP_001-92

Fig. 10. The corresponding records in EAV_TBL

We note that, although the information content in the two implementations is equivalent, there is some loss in flexibility of querying with the consolidated bridge implementation. There may be some advantage to having the information in separate dimensions, rather than consolidated in a single dimension.

5.3 Performance Testing and Results

We obtained two sets of performance results in order to compare the two implementations of the multiple groups scenario. In the case of multiple bridges, we fully processed the cube initially with the first three pairs of bridge tables/outer dimension tables.

Once the cube had been initialised and fully processed in this way, we added in the bridge table/outer dimension table pairs, one pair at a time. After the addition of each successive pair, we re-processed the cube, measuring the CPU time and the duration at each re-processing. The re-processing was explicitly requested to be the minimum needed to accommodate the changed design of the multidimensional database into the existing cube structure. In this way, we obtained cube processing performance information for seven data points, i.e. where there were 4, 5, 6, 7, 8, 9 and 10 bridge table/outer dimension pairs in the design.

For the consolidated bridge implementation, instead of changing the design of the multidimensional database by successively adding bridge table/outer dimension pairs, the equivalent information was inserted into BRIDGE_TBL and EAV_TBL. The information from the first three pairs was added to BRIDGE_TBL and EAV_TBL then the cube was fully processed. After this, the information in each successive pair was inserted into BRIDGE_TBL and EAV_TBL and, after each such insertion, the cube was re-processed.

Again, we obtained cube processing performance information for seven data points. At each such point, the information content of BRIDGE_TBL and EAV_TBL was the same as the information content of the multiple bridges implementation.

Table 2. Performance results

Size in bridges	Multi-Bridge Duration (microseconds)	Multi-Bridge CPU Time (milliseconds)	Consolidated Bridge Duration (microseconds)	Consolidated Bridge CPU Time (milliseconds)
4	608	265	562	78
5	530	343	328	62
6	983	546	452	16
7	998	499	452	78
8	1435	640	328	47
9	1404	858	312	47
10	1732	874	312	78

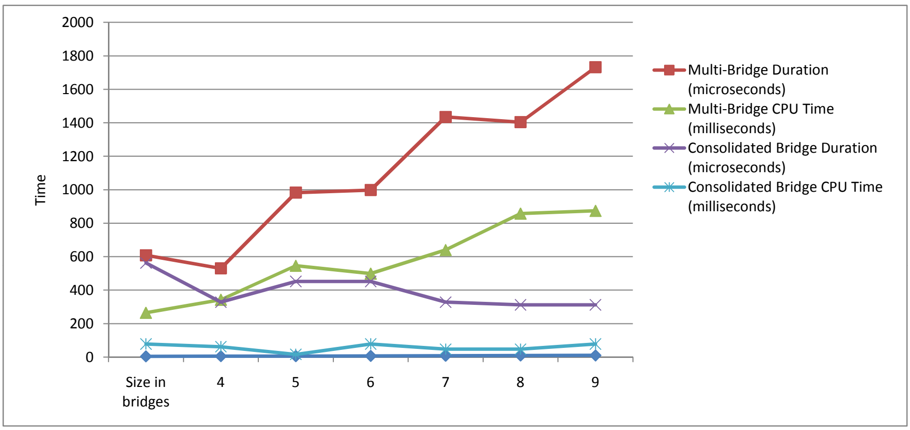


Fig. 11. Performance results

The results of the performance tests are given in Table 2 and illustrated in Figure 11.

The superior performance of the Consolidated Bridge implementation is a consequence of the fact that only minimal cube re-processing is necessary when all that happens is that new rows are inserted into existing tables in the multidimensional database from which the OLAP cube is constructed. This means that the Consolidated Bridge implementation can take advantage of the incremental processing option in SQL Server Analysis Services.

By contrast, changing the design by adding new tables to the multidimensional database requires re-processing much of the cubes existing structure. As the cube gets larger, the amount of work involved in re-processing gets bigger in proportion.

6 Conclusions

The integration of new data into a multidimensional OLAP analysis model may be a frequent, but a non-trivial task. To avoid frequent re-design and re-computation of the existing cube, we propose the use of EAV tables to integrate the new data into the existing design, at the same time avoiding sparsity. We have demonstrated the efficiency of our approach by empirical testing, comparing the use of EAV tables with a conventional approach.

Our work shows the flexibility of the use of EAV tables combined with an initial conventional design. The results show that the use of EAV tables should be considered wider in OLAP analysis. We are planning to investigate this development in more detail in the future.

Acknowledgements. This research has been partially supported by the Academy of Finland grant number 1139590.

References

1. Cameron, S.: Microsoft SQL Server 2008 Analysis Services. Microsoft Press, Redmond (2008)
2. Dinu, V., Nadkarni, P.: Guidelines for the effective use of entity-attribute-value modeling for biomedical databases. International Journal of Medical Informatics 76(11), 769–779 (2007)
3. Lee, H., Shim, J.: Pivoted table index for querying product property-value information. Journal of Software 2(5) (2010)
4. Leonhardi, B., Mitschang, B., Pulido, R., Sieb, C., Wurst, M.: Augmenting olap exploration with dynamic advanced analytics. In: Proceedings of the 13th International Conference on Extending Database Technology, EDBT 2010, pp. 687–692. ACM, New York (2010), http://doi.acm.org/10.1145/1739041.1739127
5. McBride, B.: The resource description framework (RDF) and its vocabulary description language RDFS. In: Staab, S., Studer, R. (eds.) Handbook on Ontologies. Springer, Heidelberg (2004)
6. Niedner, C.D.: The entity-attribute-value model for data and control structures in radiology. In: S/CAR 1990: Computer Applications to Assist Radiology. Symposia Foundation Press, Carlsbad (1990)
7. Niemi, T., Nummenmaa, J., Thanisch, P.: Normalising OLAP cubes for controlling sparsity. Data Knowl. Eng. 46, 317–343 (2003), http://portal.acm.org/citation.cfm?id=942051.942055
8. Russo, M.: The Many-to-Many Revolution. Electronic Book (2006), http://www.lulu.com/content/812235
9. Williams, S.: The Associative Model of Data. Lazy Software (2002)

Near Real-Time Data Warehousing with Multi-stage Trickle and Flip

Janis Zuters

University of Latvia, 19 Raina blvd.,
LV-1586 Riga, Latvia
janis.zuters@lu.lv

Abstract. A data warehouse typically is a collection of historical data designed for decision support, so it is updated from the sources periodically, mostly on a daily basis. Today's business however asks for fresher data. Real-time warehousing is one of the trends to accomplish this, but there are a number of challenges to move towards true real-time. This paper proposes 'Multi-stage Trickle & flip' methodology for data warehouse refreshment. It is based on the 'Trickle & flip' principle and extended in order to further insulate loading and querying activities, thus enabling both of them to be more efficient.

Keywords: Real-time data warehousing, data refreshment, data loading.

1 Introduction

A data warehouse (DW) is a collection of technologies aimed at enabling the knowledge worker (executive, manager, and analyst) to make better and faster decisions. Traditional online transaction processing (OLTP) systems are inappropriate for decision support. Data warehousing has become an important strategy to integrate heterogeneous data sources and to enable online analytic processing (OLAP) [1].

Traditionally a data warehouse is refreshed periodically (e.g. weekly, daily) and can be considered as a window on the past. Until recently, using periodically updated data was not a crucial issue. However, with enterprises such as e-business, stock brokering, online telecommunications, and health systems, for instance, relevant information needs to be delivered as fast as possible to knowledge workers or decision systems who rely on it to react in a near real-time manner, according to the new and most recent data captured by an organization's information system [2]. This makes supporting near real-time data warehousing a critical issue for such applications.

Today's business demands are trying to gradually eliminate the inscription for history. Technically, one of the main roles a data warehouse is to separate the analytical part of the system from the operational one, in order to satisfy heavy performance demands in both of them. Providing a data warehouse with ever fresher data makes this task more and more difficult – a frequent transportation of data from sources to data warehouse increases the collision risk between data loading and querying activities.

Real-time warehousing is an evitable stage of evolution of data warehouses. Unfortunately this involves a lot of challenges and trade-offs, especially those with regard

J. Grabis and M. Kirikova (Eds.): BIR 2011, LNBIP 90, pp. 73–82, 2011.

to state-of-the-art technologies; the most typical ones are faced when data refreshment can no longer be postponed to off-peak hours.

On the one hand, end-users of data warehouses need ever fresher data; on the other hand this involves additional requirements to hardware and possibly changes to data analysis behaviour due to frequent loadings. Because of more frequent and more complex operations related to data loading, OLAP performance could degrade dramatically.

Real-time data warehouses aim for decreasing the time it takes to make decisions and try to attain zero latency between cause and effect for that decision, closing the gap between intelligent reactive systems and systems processes. [3].

Moving towards *real-time* involves a number of issues/requirements:

- When loading data (near) continuously in real-time, there can't be any (transactional) system downtime [4];
- By loading data continuously (more frequently than daily), Query/OLAP downtimes and other inconveniences should be taken into account;
- The system becomes more complicated.
- This makes the triple trade-off for the whole system for the sake of real-time:
- Data extraction should be as lightweight as possible;
- OLAP activities shouldn't be much affected due to more frequent loading;
- Data warehouse designers/users shouldn't suffer a lot from additional complicacy of the system.

To solve this, a good deal of real-time approaches, methodologies and technologies have been proposed in each of the stages, int. al. ETL (extract-transform-load), data modelling, and data analysis.

The proposed approach of 'Multi-stage Trickle & flip' lies beyond actual data transportation issues between the source systems and the data warehouse (e.g. change data capture, CDC) and assumes conventional ETL adapted to get near real-time warehousing put into effect.

2 Related Work

In general, the principle of 'real-time' is approached in several ways – from technologies of data transportation to DW architectures and OLAP query issues.

One of the focuses is that of true real-time data transportation between the operational systems and the data warehouse. Such technologies, approaches and protocols, concerning data transport and integration, lie beyond the scope of this paper. Here well matured ETL systems are assumed, and the described approaches address the aspects of DW affecting structures and manipulations over the data after they have been loaded from the sources.

Traditionally an ETL system (and thus DW refreshment, in general) works in a *batch mode*, but *continuous ETL* technologies are also being evolved.

Near real-time data warehousing addresses the challenge of need for fresh data by simply shortening the data warehouse refreshment intervals and hence, delivering source data to the data warehouse with lower latency [5]. This approach is referred to as near real-time data warehousing or *microbatch* ETL [6]. In contrast to "true" real-time solutions this approach builds on the mature and proven ETL system and does not require the re-implementation of the transformation logic.

A workable solution to it is *near real-time ETL*, where the loading frequencies are simply increased without any other changes on the system (Fig. 1). This is the cheapest and the easiest way to solve the problem and is a good choice for relatively small data warehouses until off-peak hour loadings become an issue.

As refreshment schedule is changed, additional side effects appear. One is that of refreshment anomalies during the ETL process, addressed in [5].

If the refreshment rate exceeds microbatch basis and loading is going to become continuous, this will require new approaches and technologies lying beyond the scope of this paper.

The 'Trickle and Flip' approach helps avert the scalability issues associated with querying tables of a data warehouse that are being simultaneously updated. [4] Here, instead of continuous (or very frequent microbatch) loading of data directly into warehouse tables a staging area is used. To make things easier, the staging tables are exactly of the same format as the target tables. Applying the principle of 'Trickle & Flip', on a periodic basis the staging tables are duplicated and the copy is swapped with the data warehouse tables. For smaller data warehouses the staging tables can contain a complete copy of all data while for bigger ones a typical case would be the data for the current day being put into a separate partition.

[3] proposes an integrated data warehouses loading methodology that covers as many as four different areas of operation: (a) data warehouse schema adaptation, (b) ETL loading procedures, (c) OLAP query adaptation, and (d) DW database packing and reoptimization. As previously, here also the principle is used to have the same format for temporary tables as of the DW.

3 Trickle and Flip

This Section goes deeper into the 'Trickle & Flip' approach, as further in this paper this particular principle is being evolved.

If using 'Trickle & Flip', only very small data warehouses can do without a real-time partition (Fig. 2). Applying the principle of 'Trickle & Flip' with real-time partitions, on a periodic basis the staging tables are duplicated and the copy is swapped with the real-time tables (Figs. 3 and 4).

When used, real-time partitions imply additional requirements to the system. A real-time partition must [7]:

- Contain all the activity occurred since the last update of the static data;
- Link as seamlessly as possible to the static data tables;
- Be so lightly indexed that incoming data can be continuously fed in;
- Support high performance querying.

In particular, the latter two requirements are competing ones and thus potentially leading to awkward trade-offs or workarounds.

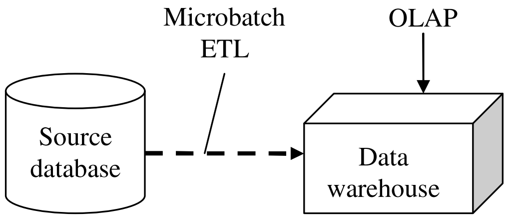


Fig. 1. Near real-time warehousing with near real-time ETL

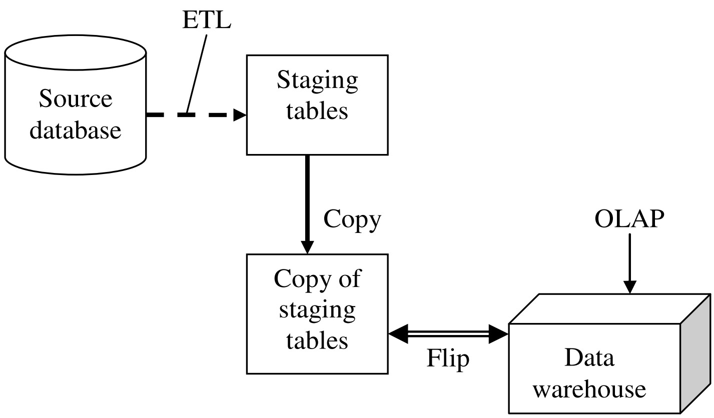


Fig. 2. Data loading using 'Trickle & Flip' with complete copy of data in the staging tables. Typical cycle times range from hourly to every minute or even faster

If the integration of real-time partition is implemented through views, the swap operation would only consist of changing the view definition. Even though flipping operation is a lightweight one, most likely it might be advisable to temporally pause the OLAP server.

In general, integrating real-time partitions into a system is very a complex task in terms of engineering. The success is largely up to ability of the query tool to encapsulate this complexity and to hide it from end-users.

A separate real-time data partition is a valuable technique to reduce loading/querying conflicts, yet at each flip (and this is to occur many times a day) it is advisable to restrict querying. If the warehouse is advancing towards real-time, the cycle times can be as frequent as of several minutes, and this would put a heavy burden to the end-users working on real-time data.

In this Section and further, it is assumed that the real-time data is loaded into the static data warehouse on a daily basis (in peak-off hours), so this particular loading is not covered by the described refreshment methodologies.

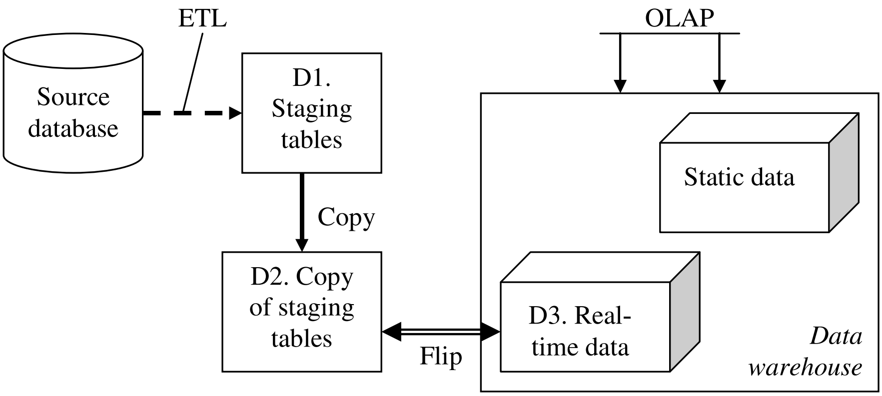


Fig. 3. Data loading using 'Trickle & Flip' with a separate real-time partition. Real-time data will typically contain the data of the current day only (A letter 'D' in D1, D2, and D3 stands for 'day'), thus all the three partitions (D1, D2, and D3) contain full data from the beginning of the current day with potentially different latencies.

```
ALGORITHM trickle_and_flip_refresh (R)
    D3 – real-time partition of the DW (Fig. 3)
    D1, D2 – staging partitions with the same data format as D3 (Fig. 3)
    R – refreshment rate (e.g., 1 hour)
    D1 is being continuously (or in a microbatch mode) fed from the source
BEGIN
    Do Every R  % e.g., every hour
        Copy D1 to D2  % i.e., do backup of D1 into D2
        Flip D2 and D3  % D3 should not be locked by querying
```

Fig. 4. DW refreshment algorithm using 'Trickle & Flip' (related to Fig. 3). In off-peak time, real-time data is added to the static DW and the staging tables are emptied.

Summary of the 'Trickle & Flip' approach (assuming the refreshment rate to be one hour):

- All the three partitions contain approximately the data from the beginning of the day up to now;
- Normally, the data of the real-time partition is not older than 1 hour;
- Each hour the process of duplicating the data of the current day is being performed;
- Each hour it is advisable to restrict querying (on real-time data)
- The issues of the 'Trickle & Flip' approach:
- For very large DWs, copying the full day data every hour could matter;
- Real-time querying being impeded every hour;
- By reducing the refreshment rate, the issues get heavier.

The ‘Multi-stage Trickle & Flip’ approach, proposed in the next Section, applies additional intermediate real-time areas in order to aver the listed drawbacks of the ‘pure’ ‘Trickle & Flip’. The main focus is put on reducing the refreshment rate to get closer to real-time.

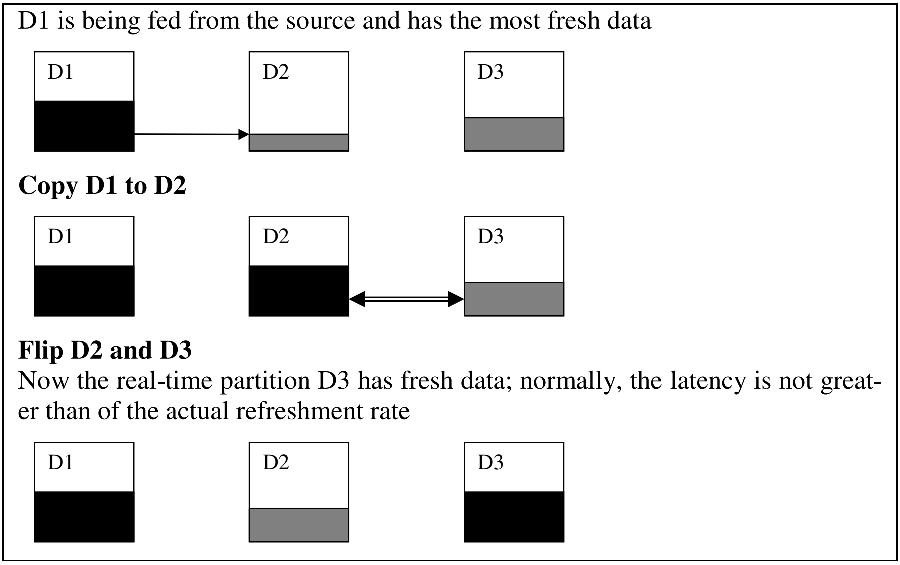


Fig. 5. An example. One iteration of a ‘Trickle & Flip’ loading (according to Fig. 4). D1 and D2 are staging areas, D3 is the real-time part of the DW.

3 Multi-stage Trickle and Flip

3.1 The Main Idea

For the pure ‘Trickle & Flip’, if the cycle times are as frequent as of several minutes, this could put a heavy burden to querying, so to the end-users working on real-time data.

To overcome this, we propose ‘Multi-stage Trickle & Flip’ approach (Fig. 6) by adding additional intermediate stages to the system, thus mitigating the competition between loading and querying processes. In particular, real-time data is divided into two (or potentially more) subpartitions, each one with a different latency (and thus, also amount of stored data).

The main objective of the new approach:

- Collisions between data loading and querying activities should be reduced.

3.2 Setup and Operation

Evolving the approach of ‘Trickle & Feed’ is based on the following assumptions:

- Adding data to a smaller table (i.e., with less data) is faster;
- Updating last changes to a table is faster than making full copy of the last version.

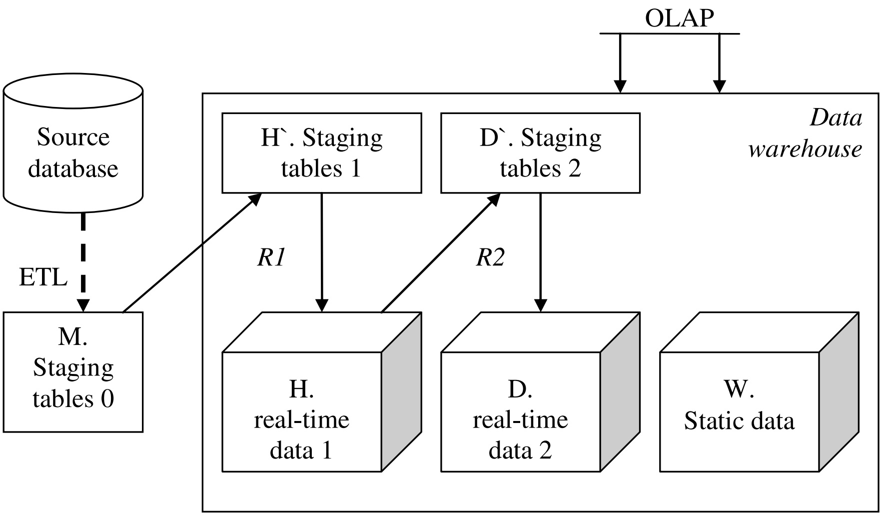


Fig. 6. Multi-stage Trickle & Flip with one intermediate level. In partition names, a letter 'M' stands for minute, 'H' stands for hour, 'D' stands for day. Data warehouse consists of partitions H, D, and W, while M, H`, and D` are staging partitions. R1 – the first stage refreshment rate, e.g., 5 minutes, R2 – the second stage refreshment rate, e.g., 1 hour.

Construction of 'Multi-stage Trickle & Flip' infrastructure (Fig. 6):

- DW is represented by static data (W) and two (or potentially more) real time partitions (D and H). Each of real time partitions is designed to contain a different degree of amount of data (e.g., of the last day or the last hour);
- The staging area M is continuously fed with the source data;
- There are two additional staging areas (for each of the real-time partitions) – D` and H` which are not affected by querying, thus always available for loading activities;
- For both stages, refreshment with two different refreshment rates is performed (See the algorithm in Fig. 7 for the first stage refreshment: the same algorithm suits second stage refreshment with parameters *R2, H, D`, D*). For simplicity of description, it is assumed that refreshment rates R1 and R2 are 5 minutes and 1 hour respectively;
- There are 3 possible levels of querying: 1) W, 2) W+D, 3) W+D+H (i.e. of latencies of 1 day, 1 hour, or 5 minutes respectively).
- An example of refreshment with this methodology is given in Fig. 8.

3.3 Benefits and Issues

The 'Multi-stage Trickle & Flip' approach reduces amount of data to be copied and possible collisions between loading and querying activities. A brief comparison between the two approaches is given in Table 1.

```
ALGORITHM multiple_trickle_and_flip_refresh (R1, M, H`, H)
    H – real-time partition for the current hour
    M, H` – staging partitions with the same data format as W (Fig. 6)
    R1 – first stage refreshment rate (e.g., 5 minutes)
    M is being continuously (or in a microbatch mode) fed from the source
BEGIN
    Do Every R1  % e.g., every 5 minutes
        Add M to H`
        Empty M
        If H is available  % not locked by querying
            Add H` to H
            Empty H`
```

Fig. 7. The first stage DW refreshment algorithm using 'Multi-stage Trickle & Flip' (related to Fig. 6). *H* will typically contain the data of the current hour (with latency of 5 minutes). *H`* will typically be empty unless *H* is not available for some time – in this case *H`* would contain one or several portions of 5 minutes data. The algorithm is visualized in Fig. 8. Exactly the same algorithm suits the second stage, but with parameters *R2, H, D`, D* respectively.

The summary obtained features of the new approach:

- Total amount of data copying is reduced;
- Collisions between data loading and querying activities have been reduced;
- By advancing the querying system (e.g., balancing needs for queries with a certain latency, separate caching for each stage) these collisions would have been more reduced;
- The approach is open to additional stages. Having appropriate querying tools available this could significantly help to approach true real-time refreshment.

The main technical issue of the methodology is vulnerability menace for query integrity if data is temporarily stored in the staging area waiting for the appropriate partition (see step #4 in Fig. 8); for the pure 'Trickle & Flip' this will simply cause higher latency.

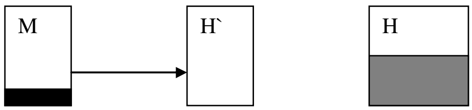
The main challenge of the approach is that of a very complicated multi-stage querying system. However the case is not "over-complex", moreover, multi-stage querying is nothing new to have come with this approach: also the pure 'Trickle & Flip' needs such in order to combine static data with real-time data. Such querying tools and methodologies are already available. [8] describes a similar principle in generating reports on a data warehouse of multiple versions (i.e., stages, in terms of this paper).

5 Conclusion

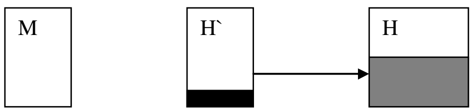
The data warehouse refreshing approach of 'Multi-stage Trickle & Flip' is proposed to mitigate the competition between the loading and querying processes when trying to approach real-time operation. It is designed as an extension to the well known principle of 'Trickle & Flip'. Although the methodology doesn't directly cover all the areas of ETL and querying, still it requires additional engineering efforts to obtain the maximum benefits, in particular, those of querying mechanism:

- Separate caching for each stage;
- A mechanism to balance needs for higher latency and the loading process.

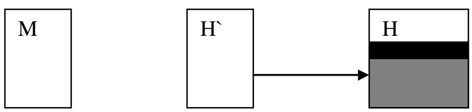
M is being fed from the source and has the freshest data of the last 5 minutes; normally, H` is empty unless H is unavailable for some time due to querying. H contains the data of the last hour:



Copy M to H`
Empty M



If H is available then copying is performed further to H
Now the real-time partition H has fresh data; normally, the latency is not greater than of 5 minutes:



If H is unavailable (due to querying activities), H` is likely to contain more then one portion of M (while waiting for H)

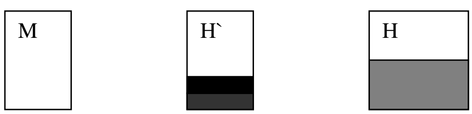


Fig. 8. An example. One iteration of a 'Multi-stage Trickle & Flip' loading; the first stage (according to the schema in Fig. 6 and the algorithm in Fig. 7). M and H` are staging areas, H is the first real-time part of the DW. The same way, the loading is performed in the second stage.

Acknowledgements. This work has been supported by ESF project No. 2009/0216/1DP/1.1.1.2.0/09/APIA/VIAA/044.

References

1. Jarke, M., Lenzerini, M., Vassiliou, Y.: Panos Vassiliadis. Fundamentals of Data Warehouses, 2nd, rev. and extended ed., XIV. Springer, Heidelberg (2003)
2. Inmon, W.H., Terdeman, R.H., Norris-Montanari, J., Meers, D.: Data Warehousing for E-Business. J. Wiley & Sons, Chichester (2001)

3. Santos, R.J., Bernardino, J.: Real-time Warehouse Loading Methodology. In: Proceedings of the 2008 International Symposium on Database Engineering & Applications (IDEAS 2008). ACM, New York (2008)
4. Langseth, J.: Real-time data warehousing: Challenges and solutions, DSSResources.COM (2004), http://dssresources.com/papers/features/langseth/langseth02082004.html
5. Jörg, T., Dessloch, S.: Near real-time data warehousing using state-of-the-art ETL tools. In: Castellanos, M., Dayal, U., Miller, R.J. (eds.) BIRTE 2009. LNBIP, vol. 41, pp. 100–117. Springer, Heidelberg (2010)
6. Kimball, R., Caserta, J.: The data warehouse ETL toolkit: Practical techniques for extracting, cleaning, conforming, and delivering data. John Wiley & Sons, Chichester (2004)
7. Kimball, R., Ross, M., Thornthwaite, W., Mundy, J., Becker, B.: The Kimball Group Reader: Relentlessly Practical Tools for Data Warehousing and Business Intelligence. John Wiley & Sons, Chichester (2010)
8. Solodovnikova, D.: Building Queries on Multiple Versions of Data Warehouse. In: Proceedings of the Eighth International Baltic Conference (DB&IS 2008), Tallinn, Estonia, pp. 75–86 (2008)

Data Consistency in Transactional Business Processes

Andreas Lodde, Antoine Schlechter, Pascal Bauler, and Fernand Feltz

Centre de Recherche Public – Gabriel Lippmann
41, rue du Brill, L-4422 Belvaux, Luxembourg
{alodde,schlecht,bauler,feltz}@lippmann.lu

Abstract. Current research about transactional workflows focuses on replicating the atomicity property for business processes using long running transactions with compensation as a replacement for rollback. But there is no standard way to guarantee data consistency between concurrent long running transactions yet. Isolation of different business processes most often is not a practical solution. To ensure reliability nevertheless, we present a generic method to detect consistency conflicts caused by the resource accesses of concurrent workflows. The method can easily be integrated into a Business Process Management System.

Keywords: Transactional business processes, long-running transactions, concurrent business processes, data consistency.

1 Introduction

More and more companies model their business processes explicitly. This modeling helps to better understand the essence in the different business processes and can help to optimize the efficiency of the business. Additionally, Business Process Management Systems (BPMS) can be used to support the personnel in following the optimized standard procedures without leaving too much room for inefficient experimentation.

Business Processes consist of sequences of activities. The failure of one activity may cause the failure of the complete business process. In such situations, we want to have some transactional processing of business processes.

The transaction concept was developed initially in the context of data processing applications in which most transactions are short running and not influenced by user interactions. Some possibilities to apply the concept of transactions on business processes were investigated in [8]. But due to the fact that most business processes are long running, especially if user interactions are involved, the basic concept of transactions is not applicable to long running business processes. Common transactions are processed according to the ACID-properties (atomicity, consistency, isolation, and durability) to achieve reliability. In most implementations, the isolation of resources causes concurrent processes to acquire access locks for all concurrently accessed resources. It is not reasonable that processes have to wait several minutes, hours or even days until a long running business process commits its transaction and thereby releases its locks on the resources. Further, the probability of deadlocks

J. Grabis and M. Kirikova (Eds.): BIR 2011, LNBIP 90, pp. 83–95, 2011.

increases with an increasing locking duration, as shown in [17]. Optimistic concurrency control as it is presented in [20] implements isolation without the use of locking. But it is not reasonable either to have concurrent transactions use outdated data for the duration of a long running business process. In fact, we do not want to isolate concurrent business processes.

From these arguments it should be clear, that we cannot simply use database transactions for the transactional processing of business processes. Each business process needs to commit its changes to the database at several times before its completion. As a consequence overall resource atomicity cannot be guaranteed within the process. In case an error occurs after such an intermediate commit, it is no longer possible to rollback the already committed changes. Research on long running transactions has shown that atomicity can be emulated by using compensation as a replacement for the rollback used in traditional transactional systems [4]. State-of-the-art Business Process Management Systems (BPMS) offer centralized and standardized compensation definition and execution capabilities. By providing the corresponding compensation operations for the different activities of a business process, the BPMS is able to carry out the compensation for the affected operations in the required sequence in case a non-recoverable failure occurs [15, 16, 18, 19].

As designated, without isolation changes on data are visible to other business processes before completion of a process. This is comparable with the use of the isolation level *Read Uncommitted* [2] in traditional transaction processing systems. It is well known that the use of this isolation level can lead to consistency conflicts such as dirty reads, non repeatable reads, or phantom reads. These errors often do not produce obvious failures. As a consequence they are often detected very late or not at all. In order to protect the business processes from this kind of unnoticed failures it is important to detect these conflicts. To date, the standard solution to this problem is to explicitly include and maintain so called dirty-flags within the business logic.

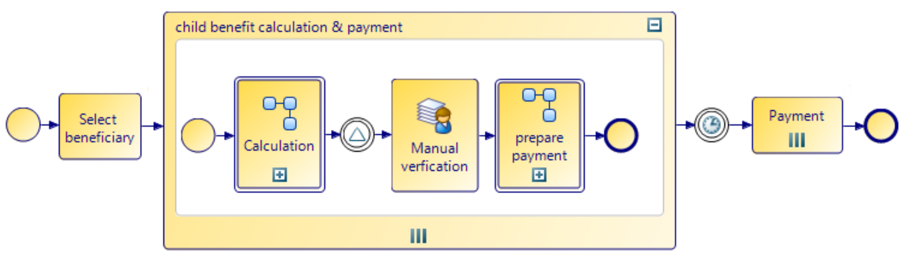


Fig. 1. Business process for the calculation and payment of child benefits

As an example for transactional business processes, we look at the calculation and payment of the child benefits to persons that are employed in Luxembourg but resident in another member country of the European Union. The main business process is shown in Fig. 1.

Due to legal constraints, it is necessary that the calculated values are manually cross-checked by the responsible persons in Luxembourg. Therefore the process has to wait until a clerk is available for carrying out the verification. The manual validation of the calculated payments has a strong effect on the overall runtime of the process. This explains why the whole process cannot be executed in a defined, short period of time.

Rather, the amount to be paid is automatically calculated at the beginning of each month using the available data. The obtained results are stored to be manually verified during the month. The actual payment is done at the end of the month.

As shown in Fig. 2, there are two concurrent processes that may update the personal data that is used in the main process. There is one process that does an automatic synchronisation of some data in the middle of each month. But in most cases personal information reaches the National Family Benefits Fund (CNPF) in paper form and has to be inserted into the system manually. This may happen at unpredictable moments. Fortunately, changes to the underlying data are rare, so that there is only a small part of the beneficiaries that needs to be recalculated and verified a second time.

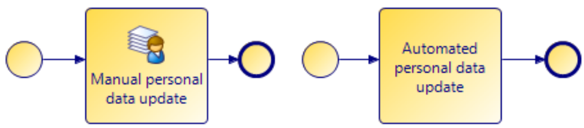


Fig. 2. Two concurrent business processes that update data used by the calculation and payment process

Changes can differ widely in their impact on the calculation and payment of the child benefit. For example, some changes may affect the fundamental entitlement of a person for child benefit or the calculated amounts to be paid. The entitlement of a person may only become active due to a change, or an active entitlement may disappear. A family benefit that should get paid may be created; an existing one may be increased, reduced or eliminated. Even if only a small subset of all payment orders change from one month to another, it is a given target to reduce the number of refunds and especially claw backs caused by obsolete data. Therefore it is necessary to calculate the family allowances by using the latest data available. Thus, we need to get notifications, when the underlying data has changed.

In this paper we will present a method to detect the use of obsolete data caused by concurrent read and write operations that can be integrated seamlessly into a Business Process Management System without affecting the underlying business logic. With this method, the timeliness of the calculated data can be guaranteed without maintaining explicit dirty flags in the business logic.

2 Consistency Conflicts

Most data conflicts can be traced back to a write access that manipulates data that has been read before by a concurrent and still running process. These consistency conflicts are called write-after-read conflicts. In case we have compensations that may invalidate data that has been written before, so called dirty reads or read-after-write conflicts may occur. Read-after-write conflicts are activated only by a data modification induced by some compensation action, and therefore they appear as write-after-read conflicts also. This is why we will primarily focus on write-after-read conflicts.

The set of all conflicts can further be divided into two subsets. The first subset concerns write operations that occur on data previously read by another process. Such a conflict is based on the concurrent modification of an element of the static set of all the data read by one process. This is why we call these conflicts *static conflicts*.

The second subset is about data that has not been read actually but should have been or the other way around. For a given query it can be determined if a row is involved in the result set or not by applying the corresponding characteristic function on each row of a table. This characteristic function corresponds to the WHERE-clause in an SQL query. To execute a query at different points in time means to reevaluate the characteristic function at these times. If some data changes between two evaluations of a characteristic function, the set of selected rows may be different even if the datasets that are contained in all of the subsequent executions remain unchanged. These conflicts may exist even if there is no static conflict. They have their origin in the dynamic changes of the set of selected rows. This is why we call these conflicts *dynamic conflicts*. Dynamic conflicts occur in business processes that do some kind of aggregation over the result set of a read operation.

In our example from Fig. 1, we first select the persons that fulfill a certain condition. Then we start calculation process for all these persons. In case personal data of the selected persons change in a way to only affect the calculations done by one of these single calculation processes, we have a static conflict. If some personal data changes in a way that unconsidered persons should actually have been considered in the creation of the calculation processes, we have a dynamic conflict.

3 A Business Process Management System with Consistency Monitoring

As mentioned before, the goal of this paper is the integration of some kind of data consistency monitoring into a business process management system. In Fig. 3 we present the architecture of such a system. Along with some modifications to the *data source* and to the *business process management system* (BPMS), we introduce a *Central Consistency Monitor* (CCM) that does all the analyses to detect conflicts. In addition, we need some *Conflict Notification and Visualization* component to publish the detected conflicts.

In order to detect conflicts, we first need to monitor all read and write operations. This is why we wrap the data source used by the business processes with a Data Access Adapter (DAA) that combines the effective data source with a *Data Access Collector*. This Data Access Collector feeds all data accesses to the *History Creation* which is responsible for saving the accesses in the *DB-Access History* for efficient use by the *Consistency Analysis*. A *Garbage Collector* will keep the DB-Access History clean of obsolete access data. The Consistency Analysis and the Garbage Collector are both triggered by the Business Process Management System. These triggers are based on the evaluation of so called *consistency regions* (CR) and *consistency check conditions* (CCC) that are defined for each business process to specify the subprocesses that need consistency monitoring and the locations where to execute the consistency tests.

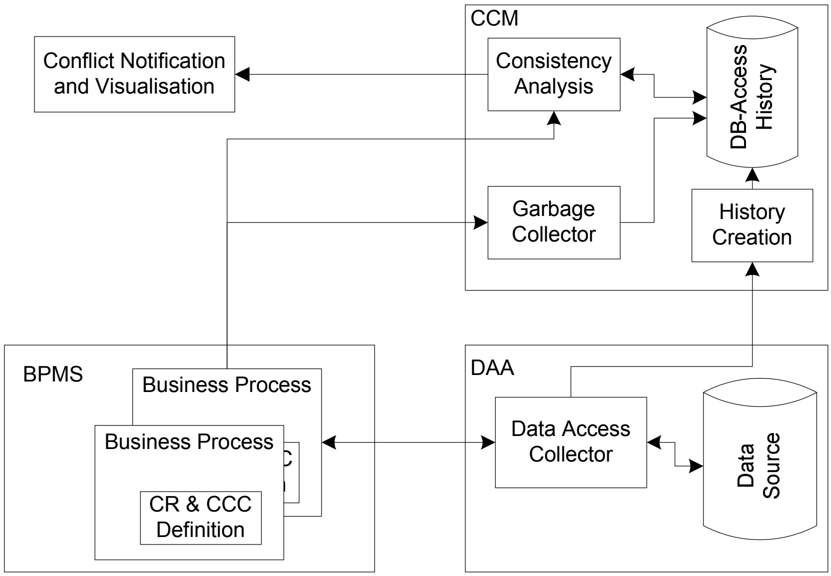


Fig. 3. Architecture of a Business Process Management System with Consistency Monitoring

3.1 Consistency Ranges and Consistency Check Conditions

When using conventional transactions we need to define the beginning and end of each transaction. In analogy, we need to define the ranges within the business processes where consistency should be monitored in order to guarantee that data inconsistencies are detected. A consistency range can range over an entire business process or only over parts of it. For example, a business process may be divided into several sequential ranges that are monitored one after the other. If such a process passes from one consistency range to another, the consistency analysis only concerns the current consistency range and not the previous one.

If a consistency range is closed, this means that it cannot be compensated any longer. As a consequence, all data that has been modified within this consistency range may be considered as valid.

After the definition of the beginning and end of a consistency range, we need to define where the actual consistency checks should be triggered. In a minimal configuration, a consistency check must be executed at the end of each consistency range and business process. But the developer might want to configure an explicit validation of previously performed operations before the execution of a very time-intensive operation.

Beside the explicit execution of consistency checks, they can also be triggered by conditions. For instance, such implicit conflict detections may be performed after every write or read operation. Another condition may be the compensation of a business process, which makes it necessary to perform a conflict analysis for all processes that have read data which were written by the compensated process. This is in fact the only situation where we may want to detect read-after-write conflicts. Some of the written data may indeed be invalidated without being modified. This is due to the conceptual difference between compensation and a real rollback.

The business process management system will use the defined consistency ranges and consistency check conditions to trigger the Consistency Analysis and the Garbage Collector in the Central Consistency Monitor.

3.2 Data Access Collector

To perform a detailed conflict analysis, it is necessary to identify datasets as accurately as possible. If the data is stored in a traditional database, we need to know what rows of a table are considered by a given operation and which column values of these rows are read or manipulated.

In addition, we need to store an identifier of the process and an identifier of the operation in order to be able to identify the exact location of a conflict. Finally, we need the timestamps of all data accesses in order to reproduce the sequence between concurrent accesses.

In most cases it is impractical to extend the existing business logic with the additional functionalities to classify resource accesses. To simplify the integration with existing systems, the data source already used by the business processes may be wrapped by a Data Access Adaptor. All calls to the resulting new data source will be routed through the data access collector to the original data source. The collector modifies the queries and extracts the relevant information from the query and the corresponding result.

Every select statement is enhanced to always return the primary key along with the desired data. In case a statement uses more than one table, the key of every table has to be requested. To obtain the keys of the rows which are manipulated by an insert, update or delete statement a corresponding select statement is generated to retrieve the keys of the affected rows and the manipulated columns.

3.3 Consistency Analysis

A conflict exists if the data used by a process is modified during its runtime by a concurrent process. In fact, if some read operation is executed at two different points in time by the same process with different results, and the process did not modify the affected datasets by itself, we have a conflict.

Thus, the simplest way to detect a conflict is to store the results of all reading operations of a process, and to compare the stored values with the values returned by a re-execution of the read operations at the time of the consistency check. This is called *conflict detection by query re-execution*.

A second possibility to detect conflicts is to record the datasets that have been read by a process. If another process performs a manipulation on exactly the same data, we have a conflict. This is called *conflict detection by read/write set comparison*.

For conflict detection by query re-execution we need to compare two complete sets of results at every test. For conflict detection by read/write set comparison we have two options of how to use this check: we can do the comparison on every write access or we can collect all write accesses and compare the complete set of writes to the set of reads. If we use hash tables to store the result sets, we can directly lookup datasets (with a computational complexity of O(1)). So, even if we choose the first option, the total effort for set comparison depends on the total number of writes that trigger a check, just the same as with the second option. The first option has the advantage to detect a conflict earlier.

Query re-execution detects all static and dynamic conflicts. On the other hand, read/write set comparison can only detect static conflicts. Query re-execution fits well with explicitly triggered consistency checks whereas read/write set comparison is best used with a trigger on each write access.

At first sight, query re-execution may look very inefficient due to the additional accesses to the database. But if the queries to the database make only a small part of the overall processing time of a business process, as for instance in most processes requiring a lot of manual user interactions, the cost for query re-execution can easily be neglected.

The real cost for query re-execution is given by the size of the result set times the number of tests plus the cost for re-execution. The real cost for read/write set comparison is given by the number of concurrent write accesses. This cost has to be balanced with the cost for fixing detected conflicts. These are given by the probabilities for the conflicts and the cost for fixing each conflict. Note that a late detected conflict is usually more expensive to fix.

3.4 Garbage Collector

As it is not reasonable to keep the information in the data access history longer than needed, business processes are able to trigger a cleanup of unneeded data access information through the garbage collector of the central consistency monitor.

At the end of a consistency range, the garbage collector may clean up all access data concerning the read operations from this consistency range. At this time, there may still exist open consistency ranges of concurrent processes that have read data which has later been modified by the closing consistency range. In addition, it is possible that the consistency checks that would find these inconsistencies have not yet been triggered. This is why we need to keep the access history of the write accesses until all concurrent open consistency ranges that have used this data before the respective modifications have closed.

On the other hand, it is also possible that a closing consistency range may have used data which has been written in a still open consistency range that could be compensated though. Consequently it is necessary to provide the option to delay the closing of a consistency range until the consistency ranges it depends on also have closed. Therefore the garbage collector manages a dependency graph that is used to close the delayed consistency ranges once all dependent consistency ranges have been closed. In addition, the graph may be used to cascade the compensation of one consistency range to the dependent consistency ranges.

3.5 Conflict Notification and Visualization

Depending on the kind of triggering condition for the analysis (a condition within the business process or an incident in a concurrent process), different parties have to be informed about the occurrence of a conflict. In general, the business process that triggers the analysis should wait for a result to react appropriately itself. The situation is different if an incident in a concurrent workflow triggers the conflict detection. In case of a compensation for example, it would be necessary to inform the processes that have read data which are no longer consistent. As it is very difficult to always react automatically on conflict notifications, our system currently only provides the detailed information about the detected conflicts. A tool to analyze the detected conflicts has been developed (Fig. 6).

4 Application

As an example, we apply the approach to the child benefit calculation and payment process from the introduction. Therefore, we only need to enhance the process model from Fig. 1 with consistency ranges and the consistency check conditions.

The complete process consists of one main global process that creates a certain number of sub-processes. As it is reasonable to relate the data used by such a sub-process only to this sub-process, it is necessary to consider sub-processes as independent business processes.

By doing so, the detection of consistency issues and compensations is simplified as only a small set of data has to be considered for each independent sub-process. If the whole process would be considered as one large process, the detection of an inconsistency in data used to calculate one child benefit would lead to the invalidation of the whole process, unless we have developed a highly complex module to determine only the correct data to invalidate.

Fig. 4 shows the independent sub-process for the calculation and verification of the child benefit for a single recipient. The process has only one consistency range over the whole process. We have defined three explicit consistency checks. All checks are performed each time the process has spent some time in a waiting state or in a time intensive operation. At these points, the probability for concurrent changes to have occurred is highest. Before the last consistency check, the process is suspended and the consistency range is kept open until the time we can assure that the underlying data will no longer change.

As a reaction to detected conflicts, we compensate the whole sub-process.

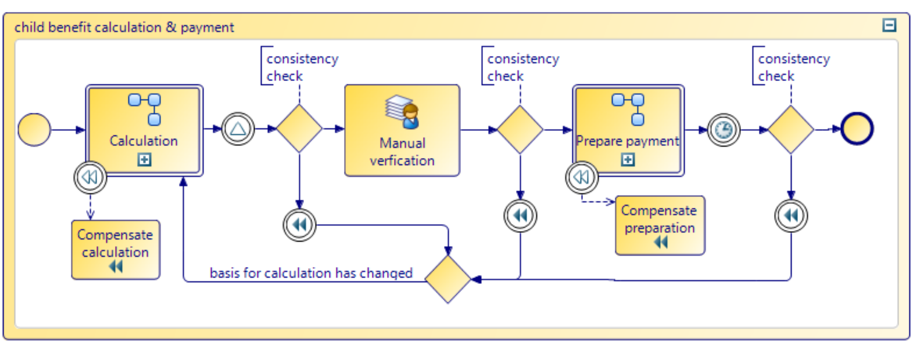


Fig. 4. Calculation and payment process of child benefits for a single beneficiary with consistency checks

These independent sub-processes are instantiated by the main process shown in Fig. 5. The main process has one consistency range. If all sub-processes have successfully completed, a last global consistency check is performed. This check is needed to verify whether the selected beneficiaries are still entitled or not. Additionally it is possible to start the child benefit calculation and payment processes for those few persons which received there formal entitlement only after the first selection and so have not yet been considered. We assume that the number of additional calculations and validations will be small, so that they can be performed in the limited remaining time before the final payment.

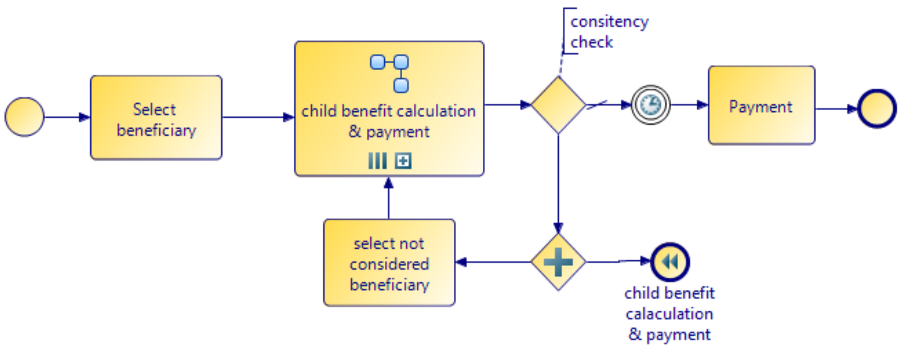


Fig. 5. Main calculation and payment process with consistency checks

The consistency checks performed in the sub-processes only need to detect static conflicts, as only modifications to the datasets actually used in the calculations are relevant. Therefore it is sufficient to use the read/write set comparison for conflict detection here. The last global consistency check has been introduced to find the dynamic conflicts on the set of selected beneficiaries. The detection of these requires the use of query re-execution.

Fig. 6 shows a simulation of the child benefit calculation and payment process. Thereby only activities and data accesses are visualized. The processes consist of one main activity which exists for the entire duration of the process and several sub-activities. All database operations of an activity are executed within traditional transactions to guarantee data consistency. The completion of a sub-activity comes along with the commit of the performed operations. For each activity we have defined a compensation method in order to undo the effects of the activity in case of an error in a later activity.

The main business process is represented at the bottom of Fig. 6. Additionally we see the child benefit calculation sub-processes for two sub-processes (260770 at the top and 260774 above the main process). Finally, there is a business process that updates the data used by the 260770 process. The conflict is visualized by the connection between the read operation R(B) of the calculation process and the update operation W(B) of the update process. As a reaction we compensate the sub-process affected by the conflict (represented by the round arrow sign). After that, we restart this sub-process after the end of the compensation.

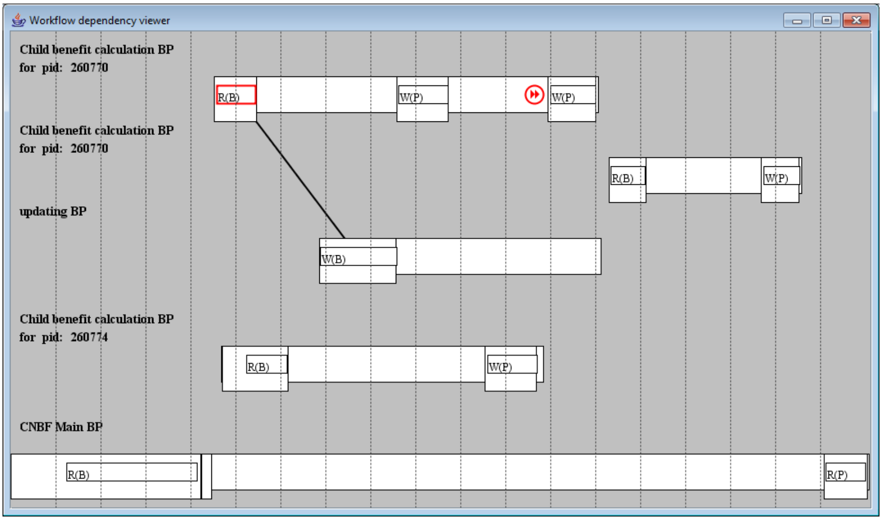


Fig. 6. Conflict Analysis and Visualization Tool

5 Related Work

Since a while, the development of a transactional model that is able to guaranty consistency in long lived transaction while avoiding locks on resources has been an important issue in research. One of the first approaches to ensure consistency in long lived transactions (LLT) has been presented in [11]. It reproduces the atomicity property for LLT by using compensation methods to undo the already committed changes to the data incase a LLT fails.

Others try to ensure consistency by using extended transaction models. In [12], different types of nested transactions are introduced and how open nested transaction can support a serialized execution with a higher concurrency than non-nested or closed nested transactions. [13] shows an improved approach for open-nested transaction, allowing data sharing between nested transaction and there parents by offering more than two levels of nesting. Transactional memory, presented in [10], ensures atomicity by encapsulating critical operations into atomic blocks. By monitoring memory accesses within atomic blocks, conflicting transactions can be detected and aborted. By adapting the idea of open nesting to transactional memory, [9] shows that open nesting is an appropriate method to increase the concurrency of transactional processes. The WS-BusinessActivity [15] specification presented by OASIS uses a variant of the open nested transaction model as the underlying model to define transactions in long-running business processes.

In order to ensure overall atomicity, the use of open-nested transactions always requires the definition of compensation actions able to undo committed results. In [7] it is shown how compensations should be orchestrated to achieve the illusion of atomicity. The correct implementation of compensation is an essential condition for using LRT. That is the reason why several research has been done on this field. In [6] the correctness of arbitrary compensating transactions in arbitrary programs is demonstrated by analysing the transaction compensations with the Conversation Calculus. Another approach to validate the behaviour of transactional service compositions is proposed in [5]. It offers the possibilities to verify the recovery mechanisms either at the design time of the transactional process or after the execution of the process to ensure service execution reliability.

These approaches do not consider consistency problems between nested transactions from different hierarchies.

In [3], a formal model is described to coordinate long running transactions in distributed systems. Thereby it is also mentioned that dependencies between service invocations can exist. By using internal and external dependency graphs the data dependencies within a single transaction or between several transactions can be expressed. [3] also addresses the need to abort all dependent transactions each time a transaction is aborted in order to maintain consistency.

A technique to discover data dependencies is presented by [1]. It uses so called deltas that represent incremental data changes to analyze data dependencies among concurrently executing processes and to determine how the failure and recovery of one process can potentially affect other processes dependent on the same data.

Our approach has been inspired by some of these ideas.

Optimistic concurrency control is about maintaining data consistency without locking. It has been investigated by [20] to increase the performance in systems where transactional conflicts are unlikely. In this approach, concurrent modifications are done on private copies of the data that are published only after a consistency validation at the commit of the transaction. In this sense, transactions are still isolated from one another, which is not viable for long running business processes. But since we do not prevent consistency conflicts either, our approach can be classified as optimistic. [14] uses an optimistic concurrency control mechanism to provide consistency for concurrent web-services. The dependencies are represented in a conflict matrix which must be provided at design time by the service developer. Our implementation dynamically creates, updates, and evaluates the data dependency graphs during the runtime of the systems in order to detect consistency conflicts as they arise.

6 Conclusions and Future Work

In this paper, we presented a method to guarantee data consistency between concurrent business processes. The method does not rely on isolation of the business processes in order to prevent consistency conflicts but it detects them as they arise. The method is based on the monitoring of consistency ranges based on consistency check conditions. Consistency ranges define the parts of the business processes that should be monitored for consistence together. The consistency check conditions define when to validate consistency. The final system consists of a central consistency monitor that is integrated into a business process management system. For the integration, the business process management systems needs to evaluate the additional information about consistency ranges and consistency check conditions during the execution of business processes. In addition, the data source used by the business processes is wrapped in order to collect the access data needed for consistency testing.

In addition to the consistency ranges and consistency check conditions, two different detection algorithms with different properties may be configured. Read/write set comparison only detects static conflicts whereas query re-execution detects all conflicts. The parameters determining the overhead for conflict detection have been explained. This information will help the developer to make good choices for the consistency monitoring configuration.

If properly configured, the overhead of the systematic consistency conflict monitoring is not much higher than the overhead caused by the equivalent explicit dirty flags in the business logic. On the other hand, there is no need for the maintenance of the dirty flag management as it is transparently included in the business process management system.

In the future, it may be possible to configure the level of detail for the data to be monitored. Additionally, we will investigate configurable automatic reaction strategies that may be executed when a consistency conflict is detected.

References

1. Urban, S.D., Xiao, Y., Blake, L., Dietrich, S.W.: Monitoring data dependencies in concurrent process execution through delta-enabled grid services. International Journal of Web and Grid Services 5, 85–106 (2009)
2. Melton, J.: ANSI X3.135-1992, American National Standard for Information Systems, Database Language SQL, American National Standards Institute (1992)
3. Moschoyiannis, S., Razavi, R., Zheng, Y., Krause, P.: Long-running transactions: semantics, schemas, implementation. In: IEEE Digital Ecosystems and Technologies. IEEE Computer Society, Los Alamitos (2008)
4. Butler, M.J., Ferreira, C.: An Operational Semantics for StAC, a Language for Modelling Long-Running Business Transactions. In: De Nicola, R., Ferrari, G.-L., Meredith, G. (eds.) COORDINATION 2004. LNCS, vol. 2949, pp. 87–104. Springer, Heidelberg (2004)
5. Gaaloul, W., Bhiri, S., Rouached, M.: Event-Based Design and Runtime Verification of Composite Service Transactional Behavior. Journal of IEEE Transactions on Services Computing 3 (2010)
6. Caires, L., Ferreira, C., Vieira, H.: A Process Calculus Analysis of Compensations. In: Trustworthy Global Computing: 4th International Symposium, Barcelona, Spain. Springer, Heidelberg (2009)
7. Butler, M., Hoare, S.T., Ferreira, C.: A Trace Semantics for Long-Running Transactions. In: Abdallah, A.E., Jones, C.B., Sanders, J.W. (eds.) Communicating Sequential Processes. LNCS, vol. 3525, pp. 133–150. Springer, Heidelberg (2005)
8. Cobb, E.E.: The impact of object technology on commercial transaction processing. VLDB Journal 6(3), 173–190 (1997)
9. Ni, Y., Menon, V.S., Adl-Tabatabai, A., Hosking, A.L., Hudson, R.L., Moss, J.E.B., Saha, B., Shpeisman, T.: Open nesting in software transactional memory. In: Proceedings of the 12th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming, San Jose, USA (2007)
10. Herlihy, M., Moss, J.E.B.: Transactional memory: architectural support for lock-free data structures. In: Proceedings of the 20th Annual International Symposium on Computer Architecture, San Diego, USA, pp. 289–300 (1993)
11. Garcia-Molina, H., Salem, K.: Sagas. In: Proceedings of the 1987 ACM SIGMOD International Conference on Management of Data, vol. 16 (1987)

12. Moss, J.E.B.: Open Nested Transactions: Semantics and Support. In: Workshop on Memory Performance Issues (2006)
13. Agrawal, K., Lee, I.A.: Sukha, J: Safe open-nested transactions through ownership. In: Proceedings of the Twentieth Annual Symposium on Parallelism in Algorithms and Architectures, Munich, Germany (2008)
14. Alrifai, M., Dolog, P., Balke, N.W.: Distributed Management of Concurrent Web Service Transactions. IEEE Transactions on Services Computing 2(4) (2009)
15. OASIS Web Service Business Activity (WS-BusinessActivity) (2007), `http://docs.oasis-open.org/ws-tx/wstx-wsba-1.1-spec.pdf`
16. BizTalk Server 2000 Business Process Orchestration 2001 B2B Protocol Standards and their Role in Semantic B2B Integration Engines. In: Bulletin of the Technical Committee on Data Engineering, vol. 24(1). IEEE Computer Society, Los Alamitos (2001)
17. Thomasian, A., Ryu, I.K.: Performance analysis of two-phase locking. IEEE Transactions on Software Engineering 17, 386–402 (1991)
18. Sun Microsystems JSR-95: J2EE Activity Service for Extended Transactions (2004), `http://jcp.org/aboutJava/communityprocess/final/jsr095/index.html`
19. Sauter, P., Melzer, I.: A Comparison of WS-BusinessActivity and BPEL4WS Long-Running Transaction. In: Proc. of Kommunikation in Verteilten Systemen, KIVS (2005)
20. Kung, H.T., Robinson, J.T.: On optimistic methods for concurrency control. ACM Trans. Database Syst., 213–226 (1981)

Business Process and Regulations: Approach to Linkage and Change Management

Peteris Rudzajs and Ilze Buksa

Department of Systems Theory and Design, Riga Technical University, Latvia
{Peteris.Rudzajs,Ilze.Buksa}@rtu.lv

Abstract. In this paper we provide a high-level architecture of the Document Analysis and Change Detection system which is used for the retrieval of regulations and document analysis and preparation for their linkage to business processes. The goal of the research is to describe approach of transformation of regulations into the form with annotated structural parts, such as chapters, sections, articles, and subarticles that enables linking of business processes to specific structural parts of the relevant regulation. We also propose approach for business process and regulation change management. By applying this approach we could ensure regulation change monitoring and thus facilitate up-to-date business process compliance to regulations.

Keywords: business process, regulation, structured document.

1 Introduction

In the scope of this paper by regulations we mean documents which record knowledge about what and how the goals of organisations should be achieved and products or services produced or provided [1], i.e., any documents which contain a valid source of regulatory requirements according to which business processes of the organisation must be executed.

At a high level of abstraction, regulations can be classified into two main groups: external regulations which are issued by the government or other superior body and organisation's internal regulations issued by the organisation itself. In this paper we do not examine other classifications of regulations and do not distinquish subtypes of regulations within classifications.

One way how to define business processes is to consider them as a series of steps designed to produce a product or service [2] which implies a strong emphasis on how work is done within the organisation [3]. In many cases business processes are related to regulations which the organization chooses or has to follow.

More and more organisations use business process management to model, simulate, execute, change, monitor, and optimize business processes. In this paper our goal is to emphasize a possibility to ensure compliance of organisation's business processes to regulations through linking business processes with regulations in business process management. Most of the business processes in an organisation are governed by the requirements of external and internal regulations. However, organisations often specify their business processes independently of the respective

J. Grabis and M. Kirikova (Eds.): BIR 2011, LNBIP 90, pp. 96–109, 2011.

regulations, even though the business processes or parts of the business processes clearly depend on the regulations. Reasons for this situation can be the lack of widely known and applicable methodologies and tool support to link business processes to regulations. The lack of linkage leads to a high risk business process non compliance to regulations and complex business process maintenance and change management. Change management plays a very important role as organisations should ensure up-to-date business processes and identify business processes which need to be changed when a regulations have been modified. Therefore in this paper we propose solution which provides linkage option at a detailed level – to link activity of the process with the structural part of the regulation.

Linking business processes with regulations in order to enable detection of change and consequent management of changes would provide:

- organisations' leadership and compliance auditors with real-time transparency and verification that business processes are running in compliance to the regulations;
- lawyers with illustration of regulations as a running business processes allowing to identify gaps in the regulations and thus optimize requirements provided in regulations, as well as allowing to see business processes that are impacted by any changes in regulations;
- software engineers with documented processes containing specified requirements of regulations according to which processes are executed, thus facilitating and speeding up the development of new systems or maintenance of the existing systems to ensure that they satisfy the requirements of the regulations.

We can assume that each business process has to satisfy requirements which are included in internal or external regulations. To manage these requirements in the business processes, it must be clear which parts of the regulations correspond to which parts of the business process. Options how a business process can be related to the regulations are as follows: (1) whole business process(es) is related to the whole regulation(s) (one-to-one, one-to-many), (2) business process step is related to the whole regulation(s), (3) business process steps are related to some part(s) (title, chapter, section, article, subarticle) of the regulation(s). Without proper tool support ensurance of such trace of relations is a challenging task.

Currently, most of the business process management tools have the function to link documents to activities of the business process or whole business process, however there is a lack of advanced tool support to enable business process linkage specifically to the structural parts of the document.

This paper is organised in four sections. Section 2 provides an analysis of related works. In Section 3 we describe the proposed Document Analysis and Change Detection (DA&CD) system. Section 4 provides an example of business process and regulation linkage in ARIS business process management tool. At the end of the paper, in Section 5, conclusions and future work are presented.

2 Related Work

Research on business process and regulation linkage and change management is related to several research domains. Hereby we are focusing on the following: (1) Linkage of business processes and regulations – the current state of solutions to manage compliance of business processes to regulations. (2) Approaches for document analysis in order to facilitate change management and linkage of business processes and regulations. (3) Change management - available solutions and approaches to the management of modifications in regulations.

2.1 Linkage of Business Processes and Regulations

Linkage of the business processes and regulations is the key to ensure business process and regulation change management. Several authors have addressed the need to ensure this link [4, 5, 6]. They have described how comprehension of regulations can be facilitated by graphical representation [1].

Araujo et al. [4] present a method for validating business processes with respect to business rules which are extracted from regulations. This method creates a link between regulations and business processes using business rules. Eijndhoven et al. [5] introduce an approach to help to achieve flexibility of the business process as regards the business rules and workflow patterns. They provide an approach based on the assumption that changes in the business process can be confined to variable isolated parts of the processes. This assumption is important since changes of some regulations most likely will impact some parts of the process, not the whole process. Governatori et al. [6] validate business processes with business contracts by providing logic based formalism for describing both the semantics of the contract and semantics of the compliance checking procedures. These approaches are promising, yet complex to implement in organizations daily practices, however they point to the necessity to link business processes and regulations.

2.2 Document Analysis

Regulations are usually represented in the form of a document. Therefore to be able to analyse the regulation we should be able to analyse the document. The aim of the document analysis is to identify (1) document's structural elements [7], e.g., chapters, sections, and articles and (2) objects used in the document [8], e.g., actors, events, rights, etc.

Ratté et al. [7] highlight the significance of identification of the document's structural elements to facilitate further analysis of the document's content (e.g., object identification in the document). Boer et al. [9] have proposed a method for extracting rights and obligations (i.e. objects) from regulations, [10] examined tool support for this and proposed a tool for semi-automatic semantic annotation of objects used in the document. Amato et al. [11] describe a system which, when given a number of regulation documents, automatically transforms them into structured documents using several ontological and linguistic knowledge levels. In this paper we propose the high-level architecture of DA&CD system that differs from the system proposed in [11] with the document retrieval module, change management module; and propose its integration with the business process management tools to link business processes to the regulation structural elements, e.g., chapters. A detailed description of the proposed DA&CD system's architecture is available in Section 3.

Identified structural elements and objects should be annotated in the source document (regulation) to further process the document and make it ready for linking to the business process. By annotation in this paper we understand the labeling of the regulation document's structural parts, such as paragraphs, chapters, sections, etc. Annotation is a form of meta-data attached to a particular section of document content [12].

Document structural element and object identification and annotation could be facilitated by using such well known text analysis software frameworks as Unstructured Information Management Architecture (UIMA) [13, 14] and General Architecture for Text Engineering (GATE) [12, 15]. In both frameworks text analysis tasks can be pipelined. It means that the results of one task could serve as input to some other task. For example, after identification of the structural parts of a regulation, e.g., sections, the document can be further processed to identify objects (actors, rights, rules etc.) in particular structural parts. We will consider the use of UIMA as text analysis framework to test its suitability for identification of structural parts of regulations (Section 3.3.2).

Our initial objective is to identify and annotate structural parts of regulation that can be referenced by the business process. Thus we gain the ability to reference a specific part of the document, e.g., section of the regulation that should be directly applied when executing a business process.

2.3 Change Detection in Regulations

Change detection in regulations is necessary to enable tracking of changes and identification of business processes affected by these changes. It means that every change in the regulation should be considered and, if necessary, respective adjustments in business processes should be performed. Regulations are usually represented in the form of documents, thus, detection of changes should also be done in documents.

There are advanced approaches that enable change-aware regulation retrieval based on eXtensible Markup Language (XML) [1, 16] and other document formats based on XML, such as MetaLex [9], and LegalXML [10]. Since regulations are mostly available in HTML and simple textual documents like MS Word documents, those advanced approaches cannot be applied. Our approach to change detection is simpler and is based on detection of changes in document content and structure. For simple change detection, available version control systems, such as Apache Subversion (SVN) [7, 17], Concurrent Versions System (CVS) [18], or Fast version control system GIT [19] could be used. We consider change detection as one of the modules of the proposed DA&CD system (see Section 3.1) and briefly describe the approach to change detection in regulations in Section 3.4.

3 DA and CD System

To enable linkage of business processes to regulations in business process management tools, we face the following challenges: (1) regulation analysis, (2) regulation change detection and application in case of changes in the source documents, and (3) linkage integration in business process management tools. In this paper we propose a high-level architecture for DA&CD system that provides modules and approaches to deal with the above mentioned challenges. In Section 3.3 we introduce a method and prototype for analyzing the structure of regulations.

In order to develop a change detection system we need to outline a change monitoring procedure. We propose a procedure which should be able to work in both directions: changes made in the regulation should indicate changes in a business process and changes made in a business process should indicate changes in regulations.

Outline of the high-level procedure to update the business process if the regulation is updated consists of the following steps:

1) Upload the new regulation (document) in the document storage. (Section 3.3)
2) Identify structural parts of the regulation (can be automated)
3) Compare the new and existing documents and identify the modified parts (Section 3.4.)
4) Indicate affected business processes and process steps (Section 3.4.)
5) Notify a person regarding changes
6) Decide on updates (always a human task)
7) Update business processes as required (it may also be possible that changes in the regulation do not require changes in the business process)
8) Create new links/delete old links between the regulation and the business process.

Outline of a high-level procedure to update the regulation if the business process is updated consists of the following steps:

1) Identify necessary changes in the business process (always a human task). User of the tool should see the links and have access to related regulations which govern the business process;
2) Decide which process parts can be and which cannot be updated. At this step a clear division between external and internal regulations must be made. Changes in the parts of the process governed by external regulations must be made with a clear focus to keep them aligned with external regulations. Process parts governed by internal regulations can be changed as required by the organisation. In that links between business processes and regulations will clearly show which regulations and which parts of the regulations are affected and must be updated;
3) Update the regulation as required by process changes (in this case, the same as with regulation changes, it may be possible that changes, in the business process do not require any changes in the regulation);
4) Create new links/delete old links between the regulation and the business process.

3.1 High-Level Architecture of DA and CD System

Developing the architecture of DA&CD system, we make the following assumptions:

1) We do not focus on all details of the system; instead we identify and describe high-level design elements and relations between elements;
2) By design elements we mean modules, components, databases, and external systems.

The architecture of DA&CD system represented in Fig. 1 consists of the following design elements (numbers in brackets indicate corresponding elements in Fig. 1):

1) Collection module (1.1) that has a document retrieval component (1.1.1)
2) Change detection module (1.2) that consists of change detection (1.2.1) and change application components (1.2.2)
3) Natural Language Processing (NLP) module (1.3) that consists of structural (1.3.1), linguistic (1.3.2), and semantic (1.3.3) analysis components
4) Storage module (1.4) that consists of databases for source document storage and versioning (1.4.2), annotated documents (1.4.1), HTML documents (1.4.3), and change logs (1.4.4)
5) Document generation module (1.5) that performs the annotated document conversion to a HTML component (1.5.1)

External components are not included into the DA&CD system, but closely linked with it are the following:

6) Sources of regulations (2): web portals or document management systems
7) Business Process Management tool (3) consisting of a component for business process linkage to regulations in the form of HTML documents (3.1) and a component for regulation change monitoring and messaging (3.2)

Typical workflow from regulation retrieval to linking them to the business process is described in the remainder of this section. In the description the numbers in brackets correspond to elements in Fig. 1, if not stated differently.

To analyse regulations, at first they should be retrieved from the storage of regulations. For external regulations, storage usually is web portals where documents are available in the HTML format; for internal regulations, internal document management systems where documents are usually available in the DOC format are used (see 2 in Fig. 1; for details see Section 3.2). Retrieval of regulations is performed by the document retrieval component (1.1.1). When a document is retrieved, text from the document is extracted and passed to the change management component (1.2.1) to be compared with an already existing document (retrieved from SVN (1.4.2) – regulation version control repository). It is worth to mention that SVN is used to store textual documents necessary for further document analysis and processing to be able to detect any changes in the source document. Regulations should be retrieved regularly to be able to identify all changes affecting business processes. If changes are detected in the retrieved document, further change detection and application mechanisms are used (for details see Section 3.4). If the regulation is retrieved for the first time, it is stored in SVN and then passed to the structural analysis component (1.3.1) for document structural element (chapter, section, and article) detection. An annotated regulation is stored in the annotated document repository (1.4.1). In order to make a regulation linkable to most of the business process management tools, it should be converted into a HTML document (1.5.1) and stored in HTML documents repository (1.4.3). Afterwards a specific structural part of the regulation (e.g., section or article) can be linked to the business process in the business process management tool (3.1). Fig. 1 also shows Linguistic (1.3.2) and Semantic (1.3.3) analysis components, but we do not consider them in this paper since we are focusing on the linkage solution only. Change application (1.2.2), change log (1.4.4), and change monitoring & messaging (3.2) components are discussed in Section 3.4.

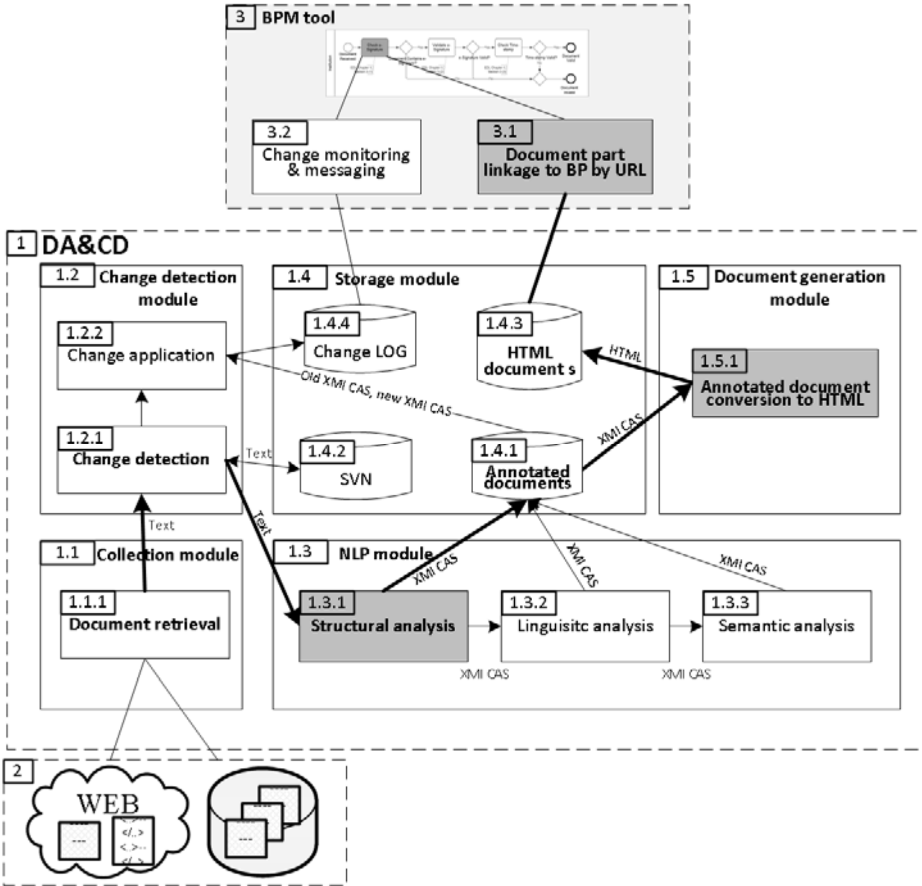


Fig. 1. Architecture of DA&CD system

3.2 Retrieval of Regulations

Retrieval of regulations (1.1.1 in Fig. 1) should be possible from various sources (2 in Fig. 1) to make them available for analysis (1.3 in Fig. 1) and for linkage to business processes (3.1 in Fig. 1). External regulations issued by the government are usually available on the web pages of governmental institutions. There are also web portals providing search facilities of regulations according to such criteria as issuer, type, subject, free text search, etc.. Storage and management of internal regulations are handled in various ways. In this paper we assume that the organisation's internal regulations are stored in document management systems. Retrieval should deal with all main sources of regulations. A web crawler can be developed to gather relevant regulations from the portals and web pages. Here we consider only direct URL's to the regulations, i.e., we are processing a selected set of regulations which have impact on the business processes being modelled.

In addition to the document's source, document's format should also be considered. Possible formats for regulations are as follows: DOC, ODT, HTML, PDF, and XML. Up to now we have employed DOC and HTML format documents.

3.3 Regulation Structural Analysis: Approach and Prototype

As mentioned in Section 2.2, in this paper we consider identification of the document's structural elements only (see also 1.3.1 in Fig. 1). As we have stated before, the solution needs to provide an option to link to the business process the whole document, as well as a specific part of the document. Usually regulations have a clear document structure per each type of the regulation. In our example we deal with Latvian law. A typical structure of the law (Fig. 2) is hierarchical and, looking top-down, a document includes introduction, title, chapter, section, article, and subarticle.

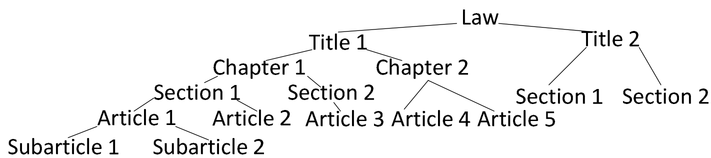


Fig. 2. Typical structure of a regulation

3.3.1 Approach

When analyzing regulations, a core element is the paragraph as each paragraph can denote typical structural elements in regulation, such as, Chapter, Section, Article, Subarticle (considered as attribute values of paragraph in this paper). To recognize and deal with the structural parts of the regulation (Fig. 2), we use the rules represented in Table 1.

Based on the structure of the regulation and structural element rules (Table 1), a special annotator was developed using UIMA (see Section 3.3.2). The aim of this annotator is to recognize the structural parts of a specific regulation.

Table 1. Rules representing each structural part of the regulation

Structural part	Rule
Chapter	If paragraph starts with Chapter, then structural part is Chapter
Section	If paragraph starts with Section, then structural part is Section
Article	If paragraph starts with a number in round brackets, e.g., (1) then it is Article
Subarticle	If paragraph starts with a number or letter following by a closing round bracket, e.g., 1) or a), then it is SubArticle
Paragraph	If a line of text contains a line end denoting symbol, then it is Paragraph

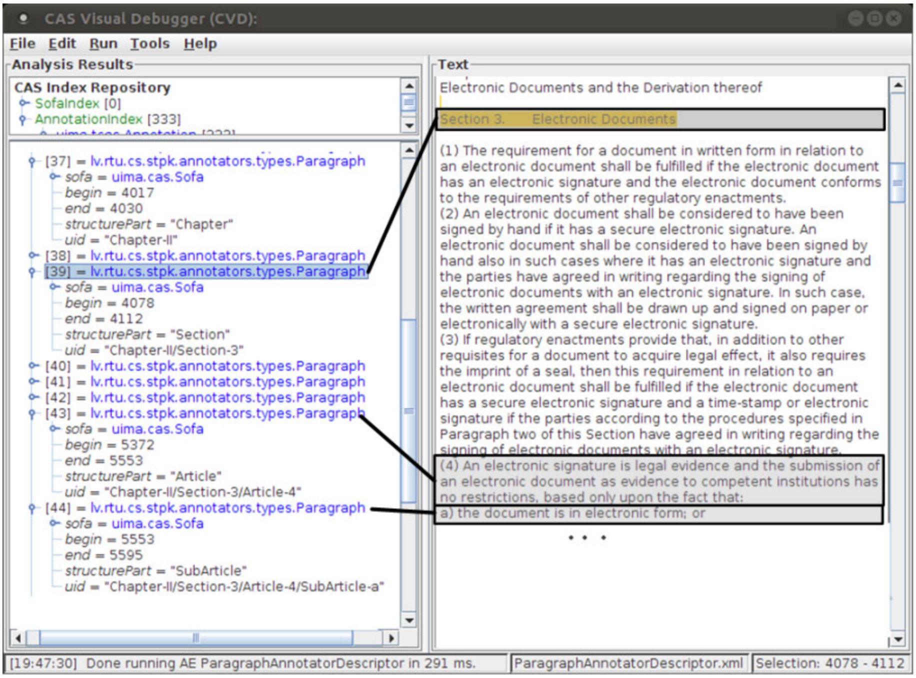


Fig. 3. Generated annotations for Electronic document law

3.3.2 Structural Analysis of Document and HTML Generation Prototype

At present we have publicly available sentence and token *UIMA* annotators. To evaluate the use of *UIMA*, we developed *Paragraph annotator*. *Paragraph annotator* is able to recognize paragraphs in the text and add additional attributes about the structural part (see Fig. 2 and Table 1) which the paragraph is representing. The developed *Paragraph annotator* was tested by *CAS* [20] *Visual Debugger*. The annotator can be packaged as *UIMA PEAR [8]* (*Processing Engine ARchive*) package and reused as a component in other text analysis systems developed using *UIMA*.

In Fig. 3 we present part of annotated Latvian *Electronic Document* law in *CAS Visual Debugger*. Every annotation has the following attributes (Fig. 3): (1) reference to *sofa* (subject of analysis - in our case it is regulation text), (2) *begin* and *end* denoting the beginning and end of the annotation, (3) *structurePart* denoting a structural part of the document such as chapter, section, and others and (4) *uid* denoting specific path to the particular subpart of the document, e.g., *Chapter-II/Section-3/Article-1*. Generated annotations are stored in *CAS XMI* document format for further analysis or processing tasks, e.g., identification of such objects as actors, roles, and events in the regulation (see 1.3.2, 1.3.3 in Fig. 1) and generation of HTML documents (see 1.5.1 and 1.4.3 in Fig. 1). Attribute *uid* is to be used for generation of a HTML document which, in turn, will be used for linkage to business processes.

An example of the generated HTML document source code is presented in Fig. 4 (HTML version visible to the end user is presented in Fig. 7, Section 4). As we can see in Fig. 4, each structural part is included in <*a*> tags with property *name* and *title*.

```
<a name="Chapter-II" title="Chapter-II"><h1>Chapter II </h1></a>
    <p>Electronic Documents and the Derivation thereof</p>
<a name="Chapter-II/Section-3" title="Chapter-II/Section-3">
    <h2>Section 3.  Electronic Documents</h2></a>
<a name="Chapter-II/Section-3/Article-1" title="Chapter-II/Section-3/Article-1">
    <p>(1) The requirement for a document in written form in relation to an
    electronic document shall be fulfilled if the electronic document has an
    electronic signature and the electronic document conforms to the
    requirements of other regulatory enactments.</p></a>
<a name="Chapter-II/Section-3/Article-2" title="Chapter-II/Section-3/Article-2">
    <p>(2) An electronic document shall be considered to have been signed by
                         . . .
```

Fig. 4. Fragment of HTML document generated from XMI CAS document

This tag serves as a bookmark in HTML document, therefore structural parts of the regulation are available for linkage in the business process management tool. It means that if a business process should be linked to Chapter-II, Section-3, then URL in business process management tool should be entered like this: http://example.com/htmlDocs/e-docLaw.*html#Chapter-II/Section-3* (***e***-docLaw.html is a document generated by regulation structural analysis and HTML generation prototype).

3.3.3 Evaluation of Annotation Results

When identification of the structural parts of the regulation is completed, we can evaluate the results of annotation. Evaluation is necessary to see the precision of the generated annotations and generated HTML document.

The prototype was tested on the document consisting of 267 structural parts. 218 structural parts were correctly recognized (*tp*) and a correct *uid* attribute was assigned (e.g., *Chapter-I/Section-3*), 45 structural parts were mislabelled (*fp*) (e.g. *Chapter-VIII/Section-26/Article-2/SubArticle-2* was mislabelled, because it would have been *Chapter-VIII/Section-26/ SubArticle-2*), and 4 structural parts were not recognized at all (*fn*). By using these numbers we calculated precision (equation 1), recall - indicates what part of all paragraphs (or parts of document's structure) were identified in the text (equation 2), and F-measure - harmonic mean value between Precision and Recall (equation 3) [3, 21]:

$$Precision = \frac{tp}{tp + fp} = 0.83 \tag{1}$$

$$Recall = \frac{tp}{tp + fn} = 0.98 \tag{2}$$

$$F\text{-}measure = 2 * \frac{precision * recall}{precision + recall} = 0.98 \tag{3}$$

The annotator recognized incorrectly mainly the cases with structure Chapter/Section/SubArticle. It suggests that the problem is in the algorithm of annotation. Taking into consideration the fact that a regulation should be structured precisely to ensure correct linkage with business processes, the precision of the annotator should be improved. We should avoid cases where a particular link to the relevant part (e.g., to article) of regulation is not available.

3.4 Change Management Mechanism

To enable business process compliance to regulations, we should ensure up-to-date business process linkage to relevant structural parts of the regulations such as Chapters, Sections, Articles, and SubArticles (Fig. 2). In the case of changes in some part of the regulation, it is necessary to detect these changes and provide mechanisms to help to identify affected business processes.

Typical regulation flow in DA&CD system starts with the document retrieval (see Fig. 1). When a document is retrieved, it is being processed by the change detection component (1.2.1 in Fig. 1) which compares the newly retrieved and existing versions of the regulation. If differences between new and existing version have been detected, structural analysis (1.3.1 in Fig. 1) of the new version should be conducted. With the aid of structural analysis we obtain a new annotated version of the regulation (in XMI CAS format). Now we have the old and the new version of the annotated regulation. Both versions are passed to Change management component (1.2.2 in Fig. 1) to identify and record the content changes of structural parts into Change Log database (1.4.4 in Fig. 1). The content of each structural part in the new version is compared to a corresponding structural part in the old version. For example, the content of Section-3/Article-1 in the new version is compared to Section-3/Article-1 in the old version. If changes are identified, then a new record is added to the Change log database. Changes in the Change log database can be tracked by the business process management tool component, namely, Change monitoring & Messaging (see 3.2 in Fig. 1). This component detects changes and identifies the linked business processes to catch attention of the business analyst. Change monitoring & messaging component is outside the proposed DA&CD system. This component can be developed as an extension of existing business process management tools.

4 Example

To test the proposed approach we have created a business process model which describes process flow for validation of an electronic document (Fig. 6). The process is modelled in ARIS business process management tool using BPMN notation. This process example describes process flow from the role perspective and does not show IT systems involved in the execution of the process.

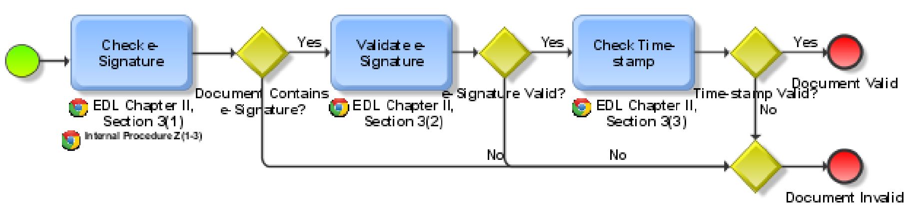
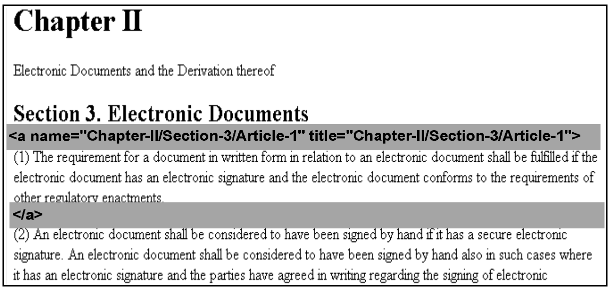


Fig. 6. Validate Document sub-process

The process must comply with the Electronic Document Law (EDL) analysed and annotated in previous sections. In the model references to the parts of the regulation are represented in the following format: Title or abbreviation of the regulation, Chapter <number>, Section <number>(<number of article>)(<number of sub-article>), for example EDL Chapter II, Section 3(1)(a). If some structural parts are not present in a particular regulation, they are excluded from the reference.

References to the structural parts of regulations are stored as attributes for activities (process steps) of the model to indicate activity specific reference; and as an attribute for the whole model, to indicate regulations applicable to the whole process.

Regulations are generated from XMI CAS document in HTML document as described in Section 3.3.2 (Fig. 7).



Chapter II

Electronic Documents and the Derivation thereof

Section 3. Electronic Documents

(1) The requirement for a document in written form in relation to an electronic document shall be fulfilled if the electronic document has an electronic signature and the electronic document conforms to the requirements of other regulatory enactments.

(2) An electronic document shall be considered to have been signed by hand if it has a secure electronic signature. An electronic document shall be considered to have been signed by hand also in such cases where it has an electronic signature and the parties have agreed in writing regarding the signing of electronic

Fig. 7. Fragment of generated HTML document from XMI CAS document

Bookmarks are created for each part of the structure, so we can have URL for each of the parts. URL is stored as the object's "Link" attributes next to the regulation reference attribute. Settings are made to see reference attributes and link attributes of the model. Now the user can click on the reference of the model and a corresponding section of the regulation will open.

5 Conclusions and Future Work

In this paper we have described approach for business process and regulation linkage and proposed a DA&CD system. Results of the approach were tested by practical example in ARIS business process management tool.

Basic concept of the proposed approach is that a regulation can be annotated and divided into structural parts thus enabling linkage options from particular business process activities to specific parts of the regulation. For the annotation of regulations an UIMA annotation was chosen. UIMA is appropriate for identification of structural parts of the regulation, because it ensures a simple development and usage mechanism of the new components (Analysis Engine, UIMA PEAR). However, the developed *Paragraph annotator* should be improved to ensure higher precision of the detection of the structural parts of regulations.

Future work includes further development and testing of this approach in order to link business processes and regulations in SAP Composition Environment. Future work also includes development of a prototype for change management approach in the context of the proposed DA&CD system. The development of change management extension for existing business process management tools should be considered, too.

Acknowledgement. This work has been supported by the European Social Fund within the project "Support for the implementation of doctoral studies at Riga Technical University" and by Accenture Ltd. and Riga Technical University in the context of research project No ZP-2010/7 "Development of linkage technology prototype for business process and normative documents bond".

References

1. Kirikova, M.: Facilitating Comprehension of Normative Documents by Graphical Representations. In: Karagiannis, D., Reimer, U. (eds.) PAKM 2002. LNCS (LNAI), vol. 2569, pp. 369–376. Springer, Heidelberg (2002)
2. Rummler, G.A., Brache, A.P.: Improving Performance – How to Manage the Whitespace on the Organization Chart, 2nd edn. Jossey-Bass Publichers, San Francisco (1995)
3. Davenport, T.H.: Process innovation; reengineering work through information technology. Harvard Business School Press, Boston (1993)
4. Araujo, B.M., Schmitz, E.A., Correa, A.L., Alencar, A.J.: A method for Validating the Compliance of Business Processes to Business Rules. In: Proceedings of the ACM Symposium on Applied Computing SAC 2010 (2010)
5. Eijndhoven, T., Iacob, M.E., Ponisio, M.L.: Achieving business process flexibility with business rules. In: Proceedings of the 12th International IEEE Enterprise Distributed Object Computing Conference EDOC 2008 (2008)
6. Governatori, G., Milosevic, Z., Sadiq, S.: Compliance checking between business processes and business contracts. In: Proceedings of the 10th International IEEE Enterprise Distributed Object Computing Conference EDOC 2006, pp. 221–232 (2006)
7. Ratté, S., Njomgue, W., Ménard, P.-A.: Highlighting Document's Structure. In: World Academy of Science, Engineering and Technology pp. 34–39 (2007)
8. Zeni, N., Kiyavitskaya, N., Cordy, J.R., Mich, L., Mylopoulos, J.: Annotating Regulations Using Cerno: An Application to Italian Documents - Extended Abstract. In: Proceedings of the 2008 Third International Conference on Availability, Reliability and Security, pp. 1437–1442. IEEE Computer Society, Los Alamitos (2008)
9. Boer, A., Winkels, R., Vitali, F.: Proposed XML Standards for Law: MetaLex and LKIF. In: Proceeding of the 2007 Conference on Legal Knowledge and Information Systems: JURIX 2007: The Twentieth Annual Conference, pp. 19–28. IOS Press, Amsterdam (2007)
10. OASIS LegalXML, `http://www.legalxml.org`
11. Amato, F., Mazzeo, A., Penta, A., Picariello, A.: Building RDF Ontologies from Semi-Structured Legal Documents. In: International Conference on Complex, Intelligent and Software Intensive Systems, CISIS 2008, pp. 997–1002 (2008)
12. ONTOTEXT, `http://www.ontotext.com/kim/getting-started/glossary#annotation`
13. UIMA PEAR, `http://uima.apache.org/doc-uima-pears.html`
14. Apache Unstructured Information Management Architecture, `http://uima.apache.org/`

15. General Architecture for Text Engineering, http://gate.ac.uk/
16. Sahilu, H., Atnafu, S.: Change-aware legal document retrieval model. In: Proceedings of the International Conference on Management of Emergent Digital EcoSystems, pp. 174–181. ACM, Bangkok (2010)
17. Apache Subversion, http://subversion.apache.org/
18. Concurrent Versions System (CVS), http://www.nongnu.org/cvs/
19. Fast version control system GIT, http://git-scm.com/
20. Common Analysis Structure (CAS), http://uima.lti.cs.cmu.edu:8080/UCR/pages/static/glossary.html
21. Olson, D.L., Delen, D.: Advanced Data Mining Techniques. Springer Publishing Company, Incorporated, Heidelberg (2008)

Parallel Tabu Search Algorithm for Data Structure Composition

Eduard Babkin, Margarita Karpunina, and Natalia Aseeva

National Research University – Higher School of Economics
Dept. of Information Systems and Technologies,
Bol. Pecherskaya 25,
603155 Nizhny Novgorod, Russia
{eababkin,naseeva}@hse.ru, karpunina-margarita@yandex.ru

Abstract. In this paper we propose a parallel tabu search algorithm to solve the problem of the distributed database optimal logical structure synthesis. We provide a reader with information about the performance metrics of our parallel algorithm and the quality of the solutions obtained in comparison with the earlier developed consecutive algorithm and other methods.

Keywords: Neural networks, tabu search, parallel programming, distributed databases.

1 Introduction

The problems of optimal composition of complex data structures play an extremely important role in many critical applications varying from cloud computing to distributed databases (DDB) [1]. In later class of applications that problem is usually formulated as synthesis of optimal logical structure (OLS). In accordance with [2] it consists of two stages. The first stage is a composition of logical record (LR) types from a set of atomic data elements (DE). The second stage is irredundant allocation of LR types among nodes in the computing network (CN). For each stage various domain-specific constraints are introduced as well as optimum criteria are specified.

From mathematical point of view the specified problem is a NP-complete non-linear optimization problem of integer programming. So far different task-specific approaches were proposed such as branch-and-bound method with a set of heuristics (BBM) [3], probabilistic algorithms, etc. However not many of them exploit benefits of parallel processing and grid technologies [4], [5], [6], [7].

In previous works the authors developed the exact mathematical formalization of the OLS problem in terms of neural networks and offered the sequential modified Tabu Search (TS) algorithm which used as a state transition mechanism by different Tabu Machines (TM) for each stage of the solution [8], [9]. The constructed algorithm produced solutions with good quality, but it was computationally efficient for small mock-up problems only.

In the present article we propose a new distributed model of TM (DTM) and a computationally efficient parallel algorithm for solution of complex OLS problems.

J. Grabis and M. Kirikova (Eds.): BIR 2011, LNBIP 90, pp. 110–123, 2011.

The article has the following structure. In Section 2 we outline critical elements of TM. For the sake of consistency Section 3 briefly presents formal definition of OLS problem and our previous proposals of that problem mapping onto the consecutive TM. In Section 4 general description of newly proposed DTM-algorithm is given and Section 5 specifies it in details. Section 6 describes evaluation of the proposed parallel algorithm. Overview of the results in Section 7 concludes the article.

2 Short Overview of Tabu Machine Model and Dynamics

In our work we use the generic model of TM as it was specified by Minghe Sun and Hamid R. Nemati [10] with the following important constituents.

$S = \{s_1,...,s_n\}$ is the current state of the TM, it is collectively determined by the states of its nodes.

$S_0 = \{s_1^0,...,s_n^0\}$ is the state of the TM with the minimum energy among all states which are obtained by the current moment within the local period (or within the short term memory process (STMP)).

$S_{00} = \{s_1^{00},...,s_n^{00}\}$ is the state of the TM with the minimum energy among all states which are obtained by the current moment (within both the STMP and the long term memory process (LTMP)).

$T = \{t_1,...,t_n\}$ is a vector to check the tabu condition.

$E(S)$ is the TM energy corresponding to the state S.

$E(S_0)$ is the TM energy corresponding to the state S_0.

$E(S_{00})$ is the TM energy corresponding to the state S_{00}.

k is the number of iterations (i.e. the number of neural network (NN) transitions from the one state to another) from the outset of the TM functioning.

h is the number of iterations from the last renewal the value of $E(S_0)$ within the STMP.

c is the number of the LTMPs carried out by the current moment.

The following variables stand as parameters of the TM-algorithm:

l is the tabu size,
β is the parameter determining the termination criterion of the STMP,
C is a maximum number of the available LTMPs inside the TM-algorithm.

The state transition mechanism of the TM is governed by TS and performed until the predefined stopping rule is satisfied. Let's name this sequence of state transitions as a work period of the TM. It is advisable to run the TM for several work periods. It is better to begin a new work period of the TM using information taken from the previous work periods, from a "history" of the TM work by applying LTMP. In such a case TS algorithm finds a node which has not changed its state for the longest time among all neurons of the TM. And then this node is forced to switch its state.

3 A Consecutive TM-Algorithm for OLS Problem

As [2] states, the general problem of DDB OLS synthesis consists of two stages.

1. **Composition of logical record (LR) types** from data elements (DE) using the constraints on:
   - the number of elements in the LR type;
   - single elements inclusion in the LR type;
   - the required level of information safety of the system.

   In addition, LR types synthesis should take into account semantic contiguity of DEs
2. **Irredundant allocation of LR types** among the nodes in the CN using the constraints on:
   - irredundant allocation of LR types;
   - the length of the formed LR type on each host;
   - the total number of the synthesized LR types placed on each host;
   - the volume of accessible external memory of the hosts for storage of local databases;
   - the total processing time of operational queries on the hosts.

The objective of OLS synthesis is to minimize the total time needed for consecutive processing of a set of DDB users' queries. Such problem has an exact but a very large mathematical formalization. So, we provide it in the Appendix I and Appendix II of this paper due to its limited size and should refer to [2], [3], [8], [9] for further details.

In our previous work [9] we have offered a new method for formalization of the described problem in the terms of TM and have constructed TMs' energy functions as follows. TM for the first stage consists of one layer of neurons, connected by complete bidirectional links. The number of neurons in the layer is equal to I^2, where I is the number of DEs. Each neuron is supplied with two indexes corresponding to indexes of DEs and LRs. For example, $OUT_{xi} = 1$ means, that the DE x will be included to the i-th LR. All outputs OUT_{xi} of a network have a binary nature, i.e. accept values from set $\{0,1\}$. The following TM energy function for LR composition was proposed:

$$E = -\frac{1}{2} \cdot \sum_{i=1}^{I} \sum_{j=1}^{I} \sum_{x=1}^{I} \sum_{y=1}^{I} \Big[-A_1 \cdot \delta_{xy} \cdot \left(1-\delta_{ij}\right) + B_1 \cdot \delta_{ij} \cdot \left(1-\delta_{xy}\right) \cdot \left(2 \cdot a_{xy}^{g} - 1\right) - D_1 \cdot \delta_{ij} \cdot$$
$$\cdot \left(incomp\_gr_{xy} + incomp\_gr_{yx} \right) \Big] \cdot OUT_{xi} \cdot OUT_{yj} + \sum_{i=1}^{I} \sum_{x=1}^{I} \left[\frac{B_1}{2} \cdot \sum_{\substack{y=1 \\ y \neq x}}^{I} \left(a_{xy}^{g}\right)^2 + \frac{C_1}{2 \cdot F_i} \right] \cdot OUT_{xi} \quad (1)$$

Here $w_{xi,yj} = -A_1 \cdot \delta_{xy} \cdot \left(1-\delta_{ij}\right) + B_1 \cdot \delta_{ij} \cdot \left(1-\delta_{xy}\right) \cdot \left(2 \cdot a_{xy}^{g} - 1\right) - D_1 \cdot \delta_{ij} \cdot \left(incomp\_gr_{xy} + \right.$ $\left. + incomp\_gr_{yx} \right)$ are weights of neurons, $T_{xi} = \left(\frac{B_1}{2} \cdot \sum_{\substack{y=1 \\ y \neq x}}^{I} \left(a_{xy}^{g}\right)^2 + \frac{C_1}{2 \cdot F_i} \right)$ are neurons' thresholds.

The quality of logical record types composition is estimated by the value of N_q parameter defined by the following formula

$$N_q = \frac{\sum_{i=1}^{I}\sum_{x=1}^{I}\sum_{\substack{y=1\\ y\neq x}}^{I} OUT_{xi}\cdot\left(OUT_{yi}-a_{xy}^{g}\right)^2}{I\cdot(I-1)}\cdot 100\% \;. \tag{2}$$

The best value for N_q is zero. The values of N_q parameter for mock-up problems solutions are shown in Table 1.

Table 1. The values of N_q parameter (%) for mock-up problems solutions

	$I=10$	$I=20$	$I=40$
	BBM		
	15,6	36,3	18,3
	NN-GA-algorithm		
	2,2	31,6	16,3
	TM-algorithm		
Maximum %	35,6	36,3	12,6
Average %	16,1	34,8	12,5
Minimum %	4,4	28,4	12,2

For the second stage of irredundant LR allocation we offered TM with the same structure as TM for LR composition, but the number of neurons in the layer is equal to $T\cdot R_0$, where T is the number of LRs, synthesized during LR composition, R_0 is the number of the hosts available for LR allocation.

As a result of constraints translation into the terms of TM the following TM energy function for the LR allocation was obtained:

$$E=-\frac{1}{2}\cdot\sum_{r_1=1}^{R_0}\sum_{r_2=1}^{R_0}\sum_{t_1=1}^{T}\sum_{t_2=1}^{T}\left[-A_2\cdot\delta_{t_1t_2}\cdot\left(1-\delta_{r_1r_2}\right)\right]\cdot OUT_{t_1r_1}\cdot OUT_{t_2r_2}+\sum_{r_1=1}^{R_0}\sum_{t_1=1}^{T}\left[\frac{B_2\cdot\psi_0}{2\cdot\theta_{t_1r_1}}\cdot\right.$$
$$\left.\cdot\sum_{i=1}^{I}\left(x_{it_1}\cdot\rho_i\right)+\frac{C_2}{2\cdot h_{r_1}}+\frac{D_2\cdot\psi_0}{2\cdot\eta_{r_1}^{EMD}}\cdot\sum_{i=1}^{I}\left(\rho_i\cdot\pi_i\cdot x_{it_1}\right)+\frac{E_2\cdot\left(t_{r_1}^{srh}+t_{r_1}\right)}{2}\cdot\sum_{p=1}^{P_0}\left(\frac{SN_{pt_1}}{T_p}\right)\right]\cdot OUT_{t_1r_1} \tag{3}$$

Here $w_{t_1r_1,t_2r_2}=-A_2\cdot\delta_{t_1t_2}\cdot\left(1-\delta_{r_1r_2}\right)$ are weights of neurons,

$$T_{t_1r_1}=\frac{B_2\cdot\psi_0}{2\cdot\theta_{t_1r_1}}\cdot\sum_{i=1}^{I}\left(x_{it_1}\cdot\rho_i\right)+\frac{C_2}{2\cdot h_{r_1}}+\frac{D_2\cdot\psi_0}{2\cdot\eta_{r_1}^{EMD}}\cdot\sum_{i=1}^{I}\left(\rho_i\cdot\pi_i\cdot x_{it_1}\right)+\frac{E_2\cdot\left(t_{r_1}^{srh}+t_{r_1}\right)}{2}\cdot\sum_{p=1}^{P_0}\left(\frac{SN_{pt_1}}{T_p}\right)$$

are neurons' thresholds. Here the I is the number of DEs, $z_{pr_1}^{t_1}=OUT_{t_1r_1}\cdot SN_{pt_1}$ and SN_{pt_1} is introduced as a normalized sum, i.e. $SN_{pt_1}=\begin{cases}1, \text{ if } \sum_{i=1}^{I} w_{pi}^{Q}x_{it_1}\geq 1\\ 0, \text{ if } \sum_{i=1}^{I} w_{pi}^{Q}x_{it_1}=0\end{cases}$, where

w_{pi}^{Q} is the matrix of dimension $(P_0 \times I)$. That matrix shows which DEs are used during processing of different queries.

In [9] we also compared the developed TM-algorithm with other methods like [10] to estimate an opportunities and advantages of TS over our earlier approaches based on Hopfield Networks [11] or their combination with genetic algorithms (NN-GA-algorithm) [2]. Using a high-dimensional mock-up problem we found that TM solutions overcome solutions received by NN-GA-algorithm in terms of average quality on 8,7%, and overcome the quality of solutions received by BBM on 23,6% (refer to Fig. 1). CPU time for LR composition was on average 36% less that the same spent by the Hopfield Network approach. So, our TM is able to produce considerable better solutions. However this algorithm is time consuming on high-dimensional tasks, and therefore we need to construct a parallel TM-algorithm in order to validate our approach on the high-dimensional tasks and increase the performance. Moreover, the parallel algorithm helps us to reveal the influence of the tabu parameters on the tasks' solution process and to determine the dependency between tabu parameters and characteristics of our problem in order to obtain better solutions faster.

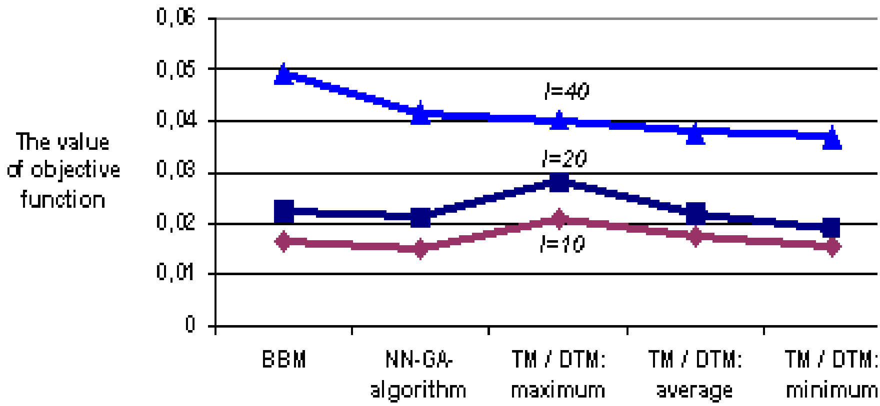


Fig. 1. The value of objective function on mock-up problems solutions

4 A General Description of DTM Functioning

The newly proposed parallel algorithm of TM exploits parallelization capabilities of the following procedures:

1. Finding a neuron to change its state;
2. Changing the value of $\Delta E(S_i)$ of neurons for using it on the next iteration;
3. Calculation of energy function value;
4. Calculation of values of auxiliary functions used in aspiration criteria of TM;
5. Transition from one local cycle to the other.

For the case of the homogeneous computational parallel cluster with multiple identical nodes the following general scheme of new parallel functionality is proposed. The set of neurons of the whole TM is distributed among all nodes' processors according to the formula $N_p = \begin{cases} n_1 + 1, \text{ if } p < n_2 \\ n_1, \text{ otherwise} \end{cases}$, where $n_1 = \left\lfloor \frac{N}{P} \right\rfloor, n_2 = N \bmod P$, N is the number of neurons in the whole TM, $p = \overline{0,(P-1)}$ is the index of processor, P is the number of processors. The number of Tabu sub-machines (TsMs) is equal to the number of available processors. So, one TsM is located on each processor and TsM with index p consists of N_p neurons. During the initialization stage neural characteristics are set to each neuron. The scheme of DTM is depicted on Fig. 2.

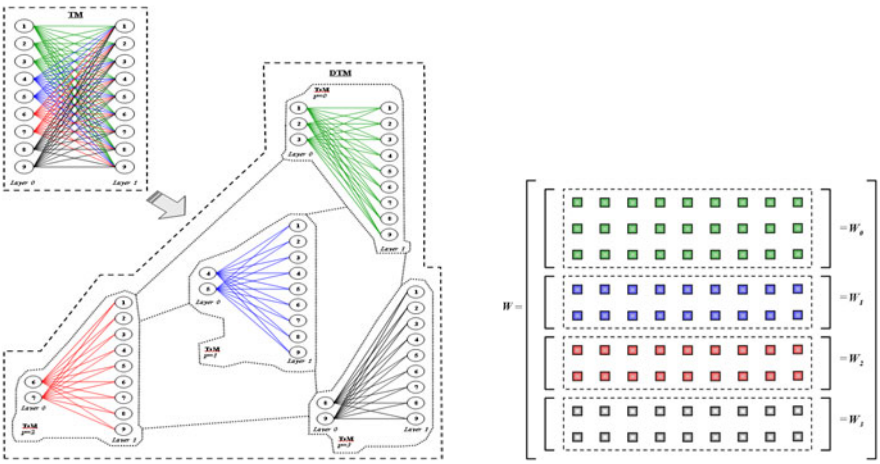


Fig. 2. DTM scheme (*left*) and method of W_p construction from the whole matrix W (*right*) for the case with $N = 9$ and $P = 4$

The same figure shows how the weight matrix of each TsM $W_p = \left\{ w_{ij}^p ; i = \overline{1, N_p}; j = \overline{1, N} \right\} = \left\{ w_{ij} ; i = \overline{\sum_{k=0}^{p-1} N_k + 1, \sum_{k=0}^{p} N_k}; j = \overline{1, N} \right\}$ is constructed from the weight matrix $W = \left\{ w_{ij} ; i, j = \overline{1, N} \right\}$ of the whole TM. When the optimal state of DTM is achieved, the results from all TsMs are united. For the proposed method of the whole TM decomposition the proposition that the energy of the whole TM is additive on the energies of TsMs including in the DTM, i.e. $E = E_0 + E_1 + \ldots + E_{P-1} = \sum_{p=0}^{P-1} E_p$, is formulated and proofed by authors but due to lack of the space is omitted in that article.

Let's consider a common implementation of DTM taking into account a parallel implementation of foregoing procedures.

Initialization. This stage assumes the construction and initialization of TsMs including into the DTM. These procedures are conducted following the mentioned above scheme of distribution of DTM neurons among the set of available processors. After the structure of each TsM is defined, TsMs are provided with the following characteristics: the matrix of neurons weights, vector of neurons thresholds, and vector of neurons biases. Thus, on the current stage we have the set of TsMs, and the elements of this set are

$$subTM_p = \{W_p, I_p, T_p, In_p\}, \quad p = \overline{0,(P-1)} \ , \tag{4}$$

where $subTM_p$ is p-th TsM, W_p is the matrix of its neurons weights, I_p is the vector of neurons biases, T_p is the vector of neurons thresholds, and In_p is the vector of initial states of TsM's neurons. Matrixes W_p and vectors I_p and T_p are defined according to the following formulas:

$$W = \{w_{ij}; i,j = \overline{1,N}\} = \begin{bmatrix} W_0 \\ W_1 \\ \vdots \\ W_{P-1} \end{bmatrix} = \begin{bmatrix} \{w_{ij}^0; i = \overline{1,N_0}; j = \overline{1,N}\} \\ \{w_{ij}^1; i = \overline{1,N_1}; j = \overline{1,N}\} \\ \vdots \\ \{w_{ij}^{P-1}; i = \overline{1,N_{P-1}}; j = \overline{1,N}\} \end{bmatrix} = \begin{bmatrix} \{w_{ij}; i = \overline{1,N_0}; j = \overline{1,N}\} \\ \{w_{ij}; i = \overline{N_0+1, N_0+N_1}; j = \overline{1,N}\} \\ \vdots \\ \left\{ w_{ij}; i = \overline{\sum_{k=0}^{P-2} N_k + 1, \sum_{k=0}^{P-1} N_k}; j = \overline{1,N} \right\} \end{bmatrix} \tag{5}$$

$$I = \{i_j; j = \overline{1,N}\} = \begin{bmatrix} I_0 \\ I_1 \\ \vdots \\ I_{P-1} \end{bmatrix} = \begin{bmatrix} \{i_j^0; j = \overline{1,N_0}\} \\ \{i_j^1; j = \overline{1,N_1}\} \\ \vdots \\ \{i_j^{P-1}; j = \overline{1,N_{P-1}}\} \end{bmatrix} = \begin{bmatrix} \{i_j; j = \overline{1,N_0}\} \\ \{i_j; j = \overline{N_0+1, N_0+N_1}\} \\ \vdots \\ \left\{ i_j; j = \overline{\sum_{k=0}^{P-2} N_k + 1, \sum_{k=0}^{P-1} N_k} \right\} \end{bmatrix} \tag{6}$$

$$T = \{t_j; j = \overline{1,N}\} = \begin{bmatrix} T_0 \\ T_1 \\ \vdots \\ T_{P-1} \end{bmatrix} = \begin{bmatrix} \{t_j^0; j = \overline{1,N_0}\} \\ \{t_j^1; j = \overline{1,N_1}\} \\ \vdots \\ \{t_j^{P-1}; j = \overline{1,N_{P-1}}\} \end{bmatrix} = \begin{bmatrix} \{t_j; j = \overline{1,N_0}\} \\ \{t_j; j = \overline{N_0+1, N_0+N_1}\} \\ \vdots \\ \left\{ t_j; j = \overline{\sum_{k=0}^{P-2} N_k + 1, \sum_{k=0}^{P-1} N_k} \right\} \end{bmatrix} \tag{7}$$

Vector In of initial states of the whole TM neurons is random generated, and then cut on P parts. Each part (i.e. In_p) corresponds to the concrete TsM.

The Local Cycle of the TM. Let's consider the local cycle of DTM.
Choose the neuron-candidate for the next move. The first step of the TM local cycle is the search of the neuron for each TsM which should change its state on the current iteration. The criterion to choose such a neuron is defined as the following:

$$\begin{gathered} \Delta E_p(S_j) = \left\{ \min\left\{ \Delta E_p(S_i) \mid i = \overline{1,N_p} \right\} \ : \ k - t_j \le l \ \vee \ E_p(S) + \Delta E_p(S_j) < E_p(S_0) \right\} . \\ p = \overline{0,(P-1)} \end{gathered} \tag{8}$$

Thus, the search of neurons satisfied to the condition (8) is performed in parallel on the hosts of CN.

The comparison of found neurons. After the neuron satisfied to the condition (8) is found on each host, the search with help of STMP reduce operations defined by authors for MPI_Allreduce function is performed within the whole DTM to find the neuron j^*, such that $\Delta E(S_{j^*}) = \min\left\{\Delta E_p(S_j) \mid p = \overline{0,(P-1)}\right\}$.

Change the energy value of neurons. After the required neuron j^* has been found, and each TsM has information about it, each neuron of $subTM_p$, $p = \overline{0,(P-1)}$ changes its $\Delta E(S_i)$ value. The calculation of DTM energy function change is done in parallel on each $subTM_p$. Further the cycle is repeated following described scheme until the condition of exit from the local cycle of the TM is satisfied.

The Global Cycle of the TM. We select neuron, that didn't change its state longest, on each TsM. The number j of this neuron on each $subTM_p$ is defined according to the following criteria:

$$\left(t_j\right)_p = \min\left\{t_i \mid i = \overline{1,N_p}\right\}, \quad p = \overline{0,(P-1)}. \tag{9}$$

The search of $\left(t_j\right)_p$ is done on the available processors in parallel according to the formula (8).

The comparison of found neurons. After the neuron satisfied to the condition (9) is found on each host, the search with help of LTMP reduce operations defined by authors for MPI_Allreduce function is performed within the whole DTM to find the neuron j^*, such that $t_{j^*} = \min\left\{\left(t_j\right)_p \mid p = \overline{0,(P-1)}\right\}$.

Change the energy value of neurons. After the required neuron j^* has been found, and each TsM has information about it, each neuron of $subTM_p$, $p = \overline{0,(P-1)}$ changes its $\Delta E(S_i)$ value. The calculation of DTM energy function change is done in parallel on each $subTM_p$. Further the cycle is repeated following described scheme until the number of LTMP calls will exceed $C : C \in Z^+, C \geq 0$ times. After that the search is stopped and the best found state is taken as the final DTM state.

5 The Algorithm of DTM Functioning

Let's try to represent the general description as an algorithm outlined step by step. We will use the following notations: N is the number of neurons in the DTM, i.e. $|S| = |S_0| = |S_{00}| = N$; N_p is the number of neurons including into the TsM $subTM_p$, where $p = \overline{0,(P-1)}$; P is the number of processors on which DTM operates.

Step 1. Construct TsMs $subTM_p$ and randomly initialize initial states of its neurons. Define the tabu-size l of DTM. Let $h = 0$ and $k = 0$ are counters of iterations in the frame of the whole DTM. Let $c = 0$, and $C \geq 0$ is the maximum number of LTMP calls in the frames of the whole DTM. Let $\beta > 0$ is defined according to inequality $\beta \cdot N > l$ in the frames of the whole DTM too.

Step 2. Find the local minimum energy state S_0. Calculate $E(S_0)$ and

$$\Delta E(S)=\begin{bmatrix}\Delta E(S_1)\\ \Delta E(S_2)\\ \vdots\\ \Delta E(S_N)\end{bmatrix}=\begin{bmatrix}\Delta E_0(S_i),\ i=\overline{1,N_0}\\ \Delta E_1(S_i),\ i=\overline{N_0+1,N_0+N_1}\\ \vdots\\ \Delta E_{P-1}(S_i),\ i=\overline{\sum_{k=0}^{P-2}N_k+1,\sum_{k=0}^{P-1}N_k}\end{bmatrix},\quad i=\overline{1,N}\ . \tag{10}$$

The values of $E_p(S_0)$ and $\Delta E_p(S_i)$ for $p=\overline{0,(P-1)}$ are calculated in parallel on P processors. Let $S_{00}=S_0$ is the best global state, and $E(S_{00})=E(S_0)$ is the global minimum of energy. Let $S=S_0$ and $E(S)=E(S_0)$. Let $t_i=-\infty,\forall i=\overline{1,N}$.

Step 3. In the frames of each $subTM_p$ choose the neuron j with $\Delta E_p(S_j)$ satisfied to $\Delta E_p(S_j)=\left\{\min\left\{\Delta E_p(S_i)\mid i=\overline{1,N_p}\right\}\ :\ k-t_j\le l\ \vee\ E_p(S)+\Delta E_p(S_j)<E_p(S_0)\right\},p=\overline{0,(P-1)}$.

Step 4. Using STMP reduce operations defined by authors, form the set $\left\{j^*,\ \Delta E(S_{j^*}),\ s_{j^*}\right\}$, where j^* is the index of neuron (in the frames of the whole DTM) changing its state at the current moment, $\Delta E(S_{j^*})$ is the change of DTM energy function value after the neuron j^* has changed its state, s_{j^*} is a new state of neuron j^*.

Step 5. If $subTM_p$ contains the neuron j^*, then $t_{j^*}=k$, $s_{j^*}=1-s_{j^*}$.

Step 6. Let $t_{j^*}=k$, $k=k+1$, $h=h+1$, $S=S_{j^*}$, $E(S)=E(S)+\Delta E(S_{j^*})$ in the frames of the whole DTM.

Step 7. Update $\Delta E(S)$ using (10). The values of $\Delta E_p(S_i)$ are calculated in parallel on P processors.

Step 8. Determine if a new state S is a new local and / or global minimum energy state: if $E(S)<E(S_0)$, then $S_0=S$, $E(S_0)=E(S)$ and $h=0$; if $E(S)<E(S_{00})$, then $S_{00}=S$ and $E(S_{00})=E(S)$ in the frames of the whole DTM.

Step 9. If $h<\beta\cdot N$, go to **Step 3.**, else go to **Step 10.**

Step 10. If $c\ge C$, then the algorithm stops. S_{00} is the best state. Else, in the frames of each $subTM_p$ choose in parallel the neuron j with $\left(t_j\right)_p$ satisfied to $\left(t_j\right)_p=\min\{t_i\mid i=\overline{1,N_p}\},\ p=\overline{0,(P-1)}$. Using LTMP reduce operations defined by authors, form the set $\left\{j^*,\ \Delta E(S_{j^*}),\ s_{j^*}\right\}$, where j^* is the index of neuron (in the frames of the whole DTM) changing its state at the current moment, $\Delta E(S_{j^*})$ is the change of DTM energy function value after the neuron j^* has changed its state, s_{j^*} is a new state of neuron j^*. Let $S_0=S_{j^*}$ and $E(S_0)=E(S)+\Delta E(S_{j^*})$, $c=c+1$ and $h=0$. Go to **Step 6.**

It's worth mentioning that on the **Step 10.** a new state of local energy minimum $E(S_0)$ is set without any auxiliary checks, i.e. it can be worse than the previous S_0. Exploiting this technique we exclude stabilization in local energy minimums and expand areas of potential solutions.

6 Performance Evaluation

In order to evaluate the performance of constructed DTM the set of experiments on mock-up problems with DTM consisting of $N = 100$, $N = 400$ and $N = 1600$ neurons were done on multi-core cluster. 372 trial solutions were obtained for each mock-up problem depending on the values of $< l, C, \beta >$ parameters of DTM.

We proposed to use an average acceleration as the metric to evaluate efficiency of DTM. The dependency of average acceleration on the number of processors for mock-up problem with $N = 1600$ is depicted on Fig. 3. DTM gives a linear acceleration.

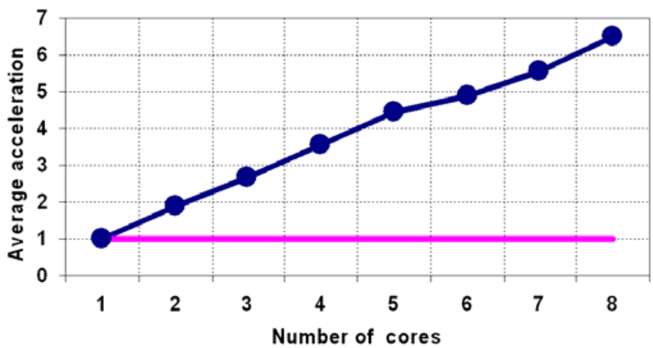


Fig. 3. Average acceleration for mock-up problem with $N = 1600$

7 Conclusion

In this paper we proposed parallel TM-algorithm for DDB OLS synthesis problem. The constructed DTM was validated and compared with the sequential TM. As expected, both approaches give the same results with the solutions quality higher than the quality of solutions received by NN-GA-algorithm [2], [8] on average 8,7% and by BBM [3] on average 23,6% on mock-up problem with higher dimension.

DTM was applied to recomposition of logical record types for the database used by human resource management tool in the international IT-company. The result set of logical record types takes the semantic contiguity of DEs on 7,5% better than the currently existing structure.

It is worth mentioning that during the DTM cycles intensive data communication between processors is carried out in the proposed algorithm. DTM provides a linear acceleration. Therefore, we can speak about the significant increasing of DTM performance in compare with its consecutive analogue for the high-dimensional problems. This statement is not contrary to our objectives, because the problem of DDB OLS synthesis is important today in view of high dimensionality.

The research was supported by Scientific Fund of NRU HSE (grant #10-04-0009).

References

1. Kant, K., Mohapatra, P.: Internet Data Centers. Computer. IEEE Computer Society, Los Alamitos (2004) 0018-9162/04
2. Babkin, E., Petrova, M.: Application of genetic algorithms to increase an overall performance of artificial neural networks in the domain of synthesis DDBs optimal structures. In: Proc. of The 5th International Conference on Perspectives in Business Informatics Research (BIR 2006), Kaunas University of Technology, Lithuania, October 6-7. Information Techonology and Control, vol. 35(3A), pp. 285–294 (2006) ISSN: 1392-124X
3. Kulba, V.V., Kovalevskiy, S.S., Kosyachenko, S.A., Sirotyuck, V.O.: Theoretical backgrounds of designing optimum structures of the distributed databases. M.: SINTEG (1999)
4. Chakrapani, J., Skorin-Kapov, J.: Massively parallel tabu search for the quadratic assignment problem. Annals of Operations Research 41, 327–341 (1993)
5. Fiechter, C.-N.: A parallel tabu search algorithm for large traveling salesman problems. Discrete Applied Mathematics 51, 243–267 (1994)
6. Garcia, B.-L., et al.: A parallel implementation of the tabu search heuristic for vehicle routing problems with time window constraints. Computers Ops. Res. 21(9), 1025–1033 (1994)
7. Porto Stella, C.S., Kitajima Joao Paulo, F.W., Ribeiro Celso, C.: Performance evaluation of a parallel tabu search task scheduling algorithm. Parallel Computing 26, 73–90 (2000)
8. Babkin, E., Karpunina, M.: Comparative study of the Tabu machine and Hopfield networks for discrete optimization problems. In: Proc. of the 14th International Conference on Information and Software Technologies, IT 2008, Kaunas, Lithuania, April 24-25, pp. 25–41 (2008), ISSN 2029-0020
9. Babkin, E., Karpunina, M.: The analysis of tabu machine parameters applied to discrete optimization problems. In: Proceedings of 2009 ACS/IEEE International Conference on Computer Systems and Aplications, AICCSA 2009, Rabat, Morocco, May 10-13, pp. 153–160. Sponsored by IEEE Computer Society, Arab Computer Society, and EMI, Morocco (2009) IEEE Catalog Number: CFP09283-CDR. ISBN: 978-1-4244-3806-8. Library of Congress: 200990028, http://www.congreso.us.es/aiccsa2009
10. Sun, M., Nemati, H.R.: Tabu Machine: A New Neural Network Solution Approach for Combinatorial Optimization Problems. Journal of Heuristics 9, 5–27 (2003)
11. Babkin, E., Karpunina, M.: About one method of DDBs structure synthesis based on the Hopfield artificial neural networks. In: Prokchorov, A.M. (ed.) Bulleten of Russian Academy of Engineering Science, Applied Mathematics and Mechanics, Moscow – Nizhny Novgorod: NSTU, vol. 12, pp. 37-46 (2005)

Appendix I: Formal Characteristics of the Subject Domain

The name	The designation
Characteristics of a DDB structure	
The set of data elements	$\mathbf{D^G} = \{d_i^g / i = \overline{1, I}\}$
The vector of elements lengths	$\boldsymbol{\rho} = \{\rho_i\}$
The vector of data element's instantiation numbers	$\boldsymbol{\pi} = \{\pi_i\}$
The matrix of a semantic contiguity of data elements	$\mathbf{A^G} = \Vert a_{ii'}^g \Vert$, where $a_{ii'}^g = 1$ if there is a semantic connection between the i-th and i'-th elements, $a_{ii'}^g = 0$ – otherwise
Characteristics of user's queries	
The set of user's queries	$\mathbf{Q} = \{q_p / p = \overline{1, P_0}\}$
The matrix of data elements usage during processing of queries	$\mathbf{W^Q} = \Vert w_{pi}^Q \Vert$, where $w_{pi}^Q = 1$ if query p uses (during processing time) the i-th element, $w_{pi}^Q = 0$ – otherwise
The matrix of frequencies of queries used by the users	$\mathbf{\Lambda^Q} = \Vert \xi_{kp}^Q \Vert$, where ξ_{kp}^Q is the frequency of the usage of the p-th query by the user k
Characteristics of the users	
The set of the users	$\mathbf{U} = \{u_k / k = \overline{1, K_0}\}$
The matrix of an attachment of the users to hosts in the computing network	$\boldsymbol{\nu} = \Vert \nu_{kr} \Vert$, where $\nu_{kr} = 1$ if the k-th user is attached to host r of computing network, $\nu_{kr} = 0$ – otherwise
The matrix of queries usage by the DDB users	$\mathbf{\Phi^Q} = \Vert \varphi_{kp}^Q \Vert$, where $\varphi_{kp}^Q = 1$ if user k uses query p, $\varphi_{kp}^Q = 0$ – otherwise
The matrix of an attachment of the queries to client hosts	$\mathbf{\Delta^Q} = \Vert \delta_{pr}^Q \Vert$, where $\delta_{pr}^Q = 1$ if $\sum_{k=1}^{k_0} \nu_{kr}\varphi_{kp}^Q \ge 1$; $\delta_{pr}^Q = 0$ if $\sum_{k=1}^{k_0} \nu_{kr}\varphi_{kp}^Q = 0$
Characteristics of the set of computing network's hosts	
The characteristics of the set of computing network's hosts	$\mathbf{N} = \{n_r / r = \overline{1, R_0}\}$
The vector of memory volumes on servers of computing network, accessible to the user	$\boldsymbol{\eta}^{\mathbf{EMD}} = \{\eta_r^{EMD}\}$, where η_r^{EMD} is the value of accessible external memory on server at host r in the computing network

The name	The designation
Average initial time characteristics	
The average time of assembly of the data block at formation of queries' data array	t^{ass}
The average time of one query formation	t^{dis}
The average time of a route choice, establishment of logic (virtual) connections between the client-host (r_1) and server-host (r_2)	$t^{ser}_{r_1 r_2}$
The average time of transfer of one data block (logical record) of query or transaction from the client-host (r_1) on the server site (r_2) on the shortest way	$t^{trf}_{r_1 r_2}$
The average time of access to LDB files and search in them of required logical records	t^{srh}
The average time of processing of one logical record on the server-host (r_2), dependent of productivity of the server	t_{r_2}

$x_{it}=1$ if the i-th data element (DE) is included into the t-th logical record (LR) type; $x_{it}=0$, otherwise.

$y_{tr}=1$ if the t-th LR type is allocated to the server of the r-th host in the computing network; $y_{tr}=0$, otherwise.

$z^{t}_{pr_2}=1$ if $\sum_{i=1}^{I} y_{tr_2} w^{Q}_{pi} x_{it} \ge 1$; $z^{t}_{pr_2}=0$ if $\sum_{i=1}^{I} y_{tr_2} w^{Q}_{pi} x_{it}=0$. Variable $z^{t}_{pr_2}$ defines types of LRs used by the *p*-th query on the server of the r_2-th host in the computing network.

$z_{pr_2}=1$ if $\sum_{t=1}^{T}\sum_{i=1}^{I} y_{tr_2} w^{Q}_{pi} x_{it} \ge 1$; $z_{pr_2}=0$ if $\sum_{t=1}^{T}\sum_{i=1}^{I} y_{tr_2} w^{Q}_{pi} x_{it}=0$. Variable z_{pr_2} defines a set of LDB server-hosts to which the p-th query addresses. T is a number of LRs types synthesizing in the solution process.

Appendix II: Mathematical Statement of the Problem

The general task of DDB OLS synthesis by criteria of a minimum of total time needed for consecutive processing of a set of DDB users' queries is formulated in the following way:

$$\min_{\{x_{it},y_{tr}\}} \sum_{k=1}^{K_0}\sum_{p=1}^{P_0} \xi^{Q}_{kp}\cdot\varphi^{Q}_{kp}\cdot\left\{\sum_{r_1=1}^{R_0} v_{kr_1}\cdot\left[\sum_{r_2=1}^{R_0} z_{pr_2}\cdot\left(t^{dis}+t^{ser}_{r_1r_2}+t^{trf}_{r_1r_2}\cdot\left(1+\sum_{t=1}^{T} z^{t}_{pr_2}\right)\right)+t^{ass}\right]+\sum_{r_2=1}^{R_0}\sum_{t=1}^{T} z^{t}_{pr_2}\cdot\left(t^{srh}_{r_2}+t_{r_2}\right)\right\}$$

subject to

1. the number of elements in the LR type $\sum_{i=1}^{I} x_{it} \le F_t, \forall t = \overline{1,T}$, where F_t is maximum number of elements in the record t;
2. single elements inclusion in the LR type $\sum_{t=1}^{T} x_{it} = 1, \forall i = \overline{1,I}$;
3. the required level of information safety of the system $x_{it} x_{i't} = 0$ for given d_i and $d_{i'}$;
4. irredundant allocation of LR types $\sum_{r=1}^{R_0} y_{tr} = 1, \forall t = \overline{1,T}$;
5. the length of the formed LR type $\sum_{i=1}^{I} x_{it} y_{tr} \rho_i \psi_0 \le \theta_{tr}, \quad \forall t = \overline{1,T}, \forall r = \overline{1,R_0}$, where θ_{tr} is the greatest allowable length of the record t determined by characteristics of the server r;
6. the total number of the synthesized LR types placed on the server r: $\sum_{t=1}^{T} y_{tr} \le h_r, \forall r = \overline{1,R_0}$, where h_r is the maximum number of the LR types supported by the local database management system of the server-host r;
7. the volume of accessible external memory of servers $\sum_{t=1}^{T} \sum_{i=1}^{I} \psi_0 \rho_i \pi_i x_{it} y_{tr} \le \eta_r^{EMD}, \forall r = \overline{1,R_0}$ for storage of local databases;
8. the total processing time of operational queries on servers $\sum_{r=1}^{R_0} \sum_{t=1}^{T} z_{pr}^{t} \cdot (t_r^{srh} + t_r) \le T_p, \forall p = \overline{1,P_0}$ for given $Q_p \in Q$, where T_p is the allowable processing time of p needed for search.

Survey on Ontology Languages

Diana Kalibatiene and Olegas Vasilecas

Vilnius Gediminas Technical University, Sauletekio al. 11, LT-10223
{diana.kalibatiene,olegas.vasilecas}@vgtu.lt

Abstract. Nowadays a number of papers are presented on the research for the ontology application for a business system modelling. For this purpose formal and executable ontologies earn a lot of attention. However, formality and executability of an ontology depends on a language, which is used to present it. This paper presents a never completely account of languages that have been used for the research community for representing ontologies. The most popular four ontology languages (KIF, OWL, RDF + RDF(S) and DAML+OIL) are reviewed. Their advantages and disadvantages are discussed. Finally, thirteen comparison criteria are distinguished and chosen ontology languages are compared. The discussion is also presented in the paper.

Keywords: Ontology, ontology language, comparison, KIF, OWL, RDF + RDF(S), DAML+OIL.

1 Introduction

In recent years, several languages have been developed to represent knowledge by ontologies. The construction of these (ontology) languages is evolving according to a layered approach, since in this layer reasoning and inference are laid. These languages must meet a number of requirements to be useful for business system modelling. According to [1], "*it must have*:

1. *a reasonably compact syntax;*
2. *a well defined semantics so that one can say precisely what is being represented;*
3. *sufficient expressive power to represent human knowledge;*
4. *an efficient, powerful, and understandable reasoning mechanism;*
5. *must be usable to build large knowledge bases."*

Moreover, an ontology language "*must describe meaning in a machine-readable way. Therefore, an ontology language needs not only to include the ability to specify vocabulary, but also the means to formally define it in such a way that it will work for automated reasoning*" [2].

Nowadays, a big emphasis is placed on web-based ontology languages. Since the web is decentralised, the languages must allow defining diverse vocabularies and letting them evolve [2].

In this paper, we discuss the most popular four ontology languages: KIF, OWL, RDF + RDF(S) and DAML+OIL. We present criteria for a comparison of ontology languages and discuss their strengths and weaknesses in relation to the chosen criteria.

J. Grabis and M. Kirikova (Eds.): BIR 2011, LNBIP 90, pp. 124–141, 2011.

The last paper is structured as follows. Section 2 presents concept of ontology. Section 3 overviews ontology languages. Section 4 presents related work on the comparison of ontology languages. Section 5 presents selecting criteria for comparison of ontology languages. And Section 6 presents the discussion on the survey obtained.

2 Concept of Ontology

Nowadays, there are a lot of definitions of ontology. It is mostly depends on the task how and why ontology is used. Classical definition, borrowed from philosophy, says that ontology means a systematic account of Existence.

According to N. Guarino [3], "*an ontology is a logical theory accounting for the intended meaning of a formal vocabulary, i. e. its ontological commitment to a particular conceptualisation of the world*". Based on such definition, an ontology consists of concepts, their definitions, relationships between concepts and constrains expressed as axioms.

However, the modern definition of ontology is extended in term of instances (in Protégé[1]-Frames ontologies) or individuals (in Ontology Web Language (OWL) [4] based ontologies), which represent objects in the domain of discourse.

According to the components, ontology defines the basic concepts, their definitions and relationships comprising the vocabulary of an application domain and the axioms for constraining interpretation of concepts and expressing complex relationships between concepts [5], [6]. Some authors, like in [7], distinguish properties from concepts also.

According to the content of a business domain knowledge, ontologies can be: *lightweight*, which describes a hierarchy of concepts related by particular relationships (e.g., is-a, part-of, etc.); *light heavyweight*, in which constraints are added to restrict the values of concepts and relationships, like cardinality constraints, possible length, etc.; and *heavyweight*, in which suitable axioms are added in order to express and restrict complex relationships between concepts and to constrain their intended interpretation.

The most existing ontologies, like WordNet[2], which can be used as a lexical ontology, Protégé ontologies (not all), ontologies presented in [8] and [9], DBpedia[3], are lightweight or light heavyweight, since those have no axioms. In heavyweight ontologies axioms defined in a framework of a description logics [10], in some kind of logic language, like KIF[4] in Protégé[1]–Frames ontology [11] and SUMO[5]. However, light heavyweight and heavyweight ontologies are the most useful for business systems modelling, since they are restricted and contain axioms and constraints, which can be used for business rules and business constraints modelling [5], [6].

---

[1] `http://protege.stanford.edu`
[2] `http://wordnet.princeton.edu/`
[3] `http://dbpedia.org/About`
[4] `http://logic.stanford.edu/kif/kif.html`
[5] `http://www.ontologyportal.org/`

Authors of this paper suggest using the definition of ontology proposed in [12] to define main criteria for ontology languages. His definition is as follows:

"***An ontology is a formal, explicit specification of a shared conceptualization***."

In this context, *conceptualization* refers to an abstract model of some phenomenon in the world, like business, that identifies that phenomenon's relevant concepts, i.e. business concepts. *Explicit* means that the type of concepts used and the constraints on their use are explicitly defined and *formal* means that the ontology should be machine understandable. Different degrees of formality are possible. Large ontologies such as WordNet[6] provide a thesaurus for over 100,000 terms explained in natural language. On the other end of the spectrum is CYC[7], which provides formal axiomating theories for many aspects of commonsense knowledge. *Shared* reflects the notion that an ontology captures consensual knowledge – it is not restricted to some individual but is accepted by a group.

3 Ontology Languages

In computer science, *ontology languages* are formal languages used to construct ontologies. They allow the encoding of knowledge about specific domains and often include reasoning rules that support the processing of that knowledge. Ontology languages are usually *declarative languages*, are almost always generalizations of *frame languages*, and are commonly based on either *first-order logic* or on *description logic* [13]. A part of existing languages are implemented into ontology development and management tools (ontology tools) or used in particular methodologies. The most interesting thing in ontology engineering is that despite the existence of many methodologies, tools and languages, it is not easy to select an appropriate ontology development technique. Sets of ontology languages, ontology tools and developing methodologies are presented in Fig. 1. You may notice that not all available methodologies, languages and tools can be selected for ontology development. This situation exists because some ontology development methodologies were built for specific ontology languages, some ontology development methodologies could be used only with certain tools. As a result, options of selecting an appropriate ontology development technique are restricted, and only certain approaches can be used for ontology development. Authors of [14] and [15] propose facilitating ontology development by constant evaluation of steps in the process of ontology development and the user guide throughout the process to solve the above problem.

According to [16], ontology languages can be divided into two categories. The first category is ***traditional ontology languages***. They are: languages based on *first-order predicate logic* (KIF, CycL), *frame-based languages* (Ontolingua, F-logic and OCML), *description logic (DL) based languages* (Loom), and other languages. The second category is Web standards, which are used to facilitate interchange on the Internet, and ontology languages, which are web standards compatible, are named ***Web-based ontology languages***. However, particular languages can be assigned to

[6] `www.cogsci.princeton.edu/~wn`
[7] `www.cyc.com`

both categories, i.e. they are based on a particular Web standard and a particular paradigm, like frames or first-order predicate logic. For example, OWL DL is based on RDF and description logic. It is presented in Fig. 2. We are going to pay more attention to the languages, which belong to both sides.

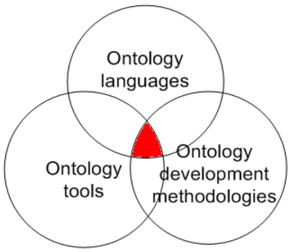


Fig. 1. Ontology development methodologies, languages and tools

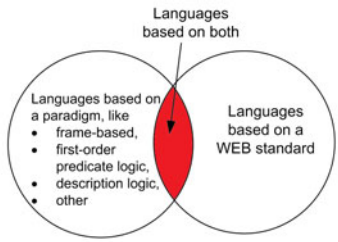


Fig. 2. Ontology languages

Below we present several ontology languages, which are going to be compared in the next section of this paper. Some advantages and disadvantages and an example are going to be presented for each chosen language.

3.1 KIF (SUO-KIF SUMO Language)

Knowledge Interchange Format (KIF)[8] is a computer-oriented language for the interchange of knowledge among disparate programs. The basis for the semantics of KIF is a *conceptualization* of the world in terms of objects and relations among those objects. Its features are as follows [17]:

- *declarative semantics* – possibility to understand the meaning of expressions without appeal to an interpreter for manipulating those expressions;
- *logically comprehensive* – the expression of arbitrary sentences in the first-order predicate calculus;
- *meta-knowledge* – the representation of knowledge about the representation of knowledge;
- the representation of *nonmonotonic reasoning rules*;
- *readability* – the definition of objects, functions, and relations;
- *implementability* – although KIF is not intended for use within programs as a representation or communication language, it should be usable for that purpose if so desired.

It was originally created in the DARPA knowledge Sharing Effort [18]. There have been a number of versions of KIF and several languages based on KIF, like SUO-KIF[9], are created. Table 1 presents an example of an ontology axiom defined in SUO-KIF: "*A rail vehicle is a vehicle designed to move on railways.*"

[8] http://logic.stanford.edu/kif/

[9] http://suo.ieee.org/SUO/KIF/suo-kif.html

Table 1. An example of SUMO axiom defined in SUO-KIF[10]

```
(subclass RailVehicle LandVehicle)
   (documentation RailVehicle
        "A Vehicle designed to move on &%Railways.")
   (=>  (instance ?X RailVehicle)
        (hasPurpose ?X
         (exists (?EV ?SURF)
           (and  (instance ?RAIL Railway)
                 (instance ?EV Transportation)
                 (holdsDuring (WhenFn ?EV)
                         (meetsSpatially ?X ?RAIL))))))
```

The main disadvantages of KIF are: a *high expressiveness* of KIF complicates the building of fully conforming systems; the resulting systems tend to be "*heavyweight*", i.e. they are larger and in some cases less efficient than systems that employ more restricted languages[7].

3.2 OWL (Lite, DL, Full)

OWL is a standard ontology language for the semantic web. It is a language extension of RDF Schema. OWL is compatible with early ontology languages, including SHOE, DAML+OIL, and provides the engineer more power to express semantics. It includes conjunction, disjunction, existentially, and universally quantified variables, which can be used to carry out logical inferences and derive knowledge. An example of an OWL is presented in Table 2. However, OWL has the following drawbacks [19]:

1. Some constructs are very *complex*, therefore three sublanguages is designed [4]:

– OWL Lite – supports a classification hierarchy and simple constraint features, like cardinality constraints. The advantage of this is that is both easier to grasp (for users) and easier to implement (for tool builders) [20]. The disadvantage is a restricted expressivity.
– OWL DL – includes all OWL language constructs with restrictions such as type separation (a class cannot also be an individual or property). It corresponds with description logics. The advantage is that it permits efficient reasoning support [20]. However, full compatibility with RDF is lost.
– OWL Full – supports maximum expressiveness and the syntactic freedom of RDF with no computational guarantees. OWL Full fully compatible with RDF, both syntactically and semantically [20]. However, it becomes undecidable and not suitable for complete reasoning in particular cases.

2. Reasoning is not efficient as there is a trade-off against time-complex cost.

– It is *not intuitive*, need to be owl-savvy to build efficient knowledge constructions.

---

[10] http://www.obitko.com/tutorials/ontologies-semantic-web/knowledge-interchange- format.html

Table 2. An example of Pizza ontology – OWL class [21]

```
<owl:Class>
    <owl:unionOf rdf:parseType="Collection">
         <owl:Class rdf:about="#MozzarellaTopping"/>
         <owl:Class
        rdf:about="#PeperoniSausageTopping"/>
         <owl:Class rdf:about="#JalapenoPepperTopping"/>
         <owl:Class rdf:about="#TomatoTopping"/>
         <owl:Class rdf:about="#HotGreenPepperTopping"/>
    </owl:unionOf>
</owl:Class>
```

3.3 RDF

The *Resource Description Framework* (RDF)[11] is a standard for describing resources on the web developed by the World Wide Web Consortium (W3C). It is designed to be read and understood by computers not displayed to people. It is suitable for describing any web resource and as such it provides interoperability between applications that exchange machine-understandable information on the web. It is written in Extensible Markup Language (XML). RDF is becoming a widely recognized language and a representation formalism that can serve as a worldwide interlingua for information interchange.

RDF is based on the idea of identifying things on the Web via URI references and expressing statements about resources in terms of simple properties and property values [22]. Each RDF statement consists of a triplet: a *subject*, a *predicate* (also called a property) and an *object*, or, as stated in some references like [2], an object, an attribute and a value. Its goal is to add formal semantics to the web and provide a data model and syntax convention for representing the semantics of the data in a standardized manner. The relationships among resources are described in terms of the named properties and values. An example of RDF is presented in Table 3.

Table 3. An example of a describing a record shop in RDF[12]

```
<rdf:Description
rdf:about="http://www.recshop.fake/cd/Hide your heart">
    <cd:artist>Bonnie Tyler</cd:artist>
    <cd:country>UK</cd:country>
    <cd:company>CBS Records</cd:company>
    <cd:price>9.90</cd:price>
    <cd:year>1988</cd:year>
</rdf:Description>
```

RDF has significant advantages over XML. They are:

- The *object-attribute structure* – all objects are independent entities.
- The *RDF model* – RDF describes an independent layer.

[11] http://www.w3.org/RDF/
[12] http://www.w3schools.com/rdf/default.asp

- An *extensible object-oriented type system* – the RDF schema (RDFS) has been introduced as a layer on the top of the basic RDF model. The RDFS can be thought of as a set of ontological modelling primitives. XML lacks this layer and some developers building a layer on the top of XML to integrate ontological primitives.
- *Developing of RDF vocabulary* – RDFS lets developers define a particular vocabulary for RDF data and specify the kind of object to which these attributes may be applied.

RDF played an important role as a basis for DARPA Agent Markup Language (DAML)[13], whose layers of logic are to be built on the top of the basic RDF framework. However, the disadvantages of RDF are: lack of affection inference mechanism and the lack of formal underlying model semantics [23].

3.4 The Ontology Interchange Language (OIL)

The *Ontology Interchange Language* (OIL) is a full-fledged Web-based ontology language based on RDFS [24]. It was created according to these requirements [25] as follows by a group of (mostly) European researchers[14] it must: a) be *highly intuitive* to the human user; b) have a *well-defined formal semantics* with established reasoning properties to ensure completeness, correctness, and efficiency; c) have a *proper link* with existing Web languages such as XML and RDF to ensure interoperability. Not all ontology languages, like CycL[15], KIF and Ontolingua[16], meet these requirements [25]. The main advantages of OIL are as follows:

- The *frame-based* and *object-oriented* modelling paradigms. OIL is based on the notion of a class and its superclasses and attributes. Relations can be defined as independent entities.
- *Well-defined formal semantics* based on DL. In addition the meaning of any expression can be described in a mathematic precise way, which enables reasoning with concept description and the automatic derivation of classification taxonomies.
- *Based on WEB standards*. OIL has a well-defined syntax in XML. It is also defined as an extension of the RDF and its extension schema (RDFS).
- It offers *different levels of complexity*. It has four layers: Core OIL, Standard OIL, Instance OIL (Standard OIL + RDFS) and Heavy OIL [25].

The following OIL expression presented in Table 4 defines *herbivore as a class, which is a subclass of animal and disjoint to all carnivores*.

However, authors of [26] and [27] find the following disadvantages of OIL. It is impossible: to define the default-value, to provide the meta-class and to support the concrete domain. Translating from OIL to RDF and back is no longer guaranteed to give an identical ontology from a modelling perspective (though semantic equivalence is still preserved).

---

[13] http://www.daml.org/
[14] http://www.ontoknowledge.org/oil
[15] http://www.opencyc.org/doc/
[16] http://ksl.stanford.edu/software/ontolingua/

Table 4. An example of a herbivore in OIL [25]

```
    <rdfs:Class rdf:ID=”herbivore”>
         <rdf:type
              rdf:resource=”http://www.ontoknowledge.org/oil/
RDFSschema/#DefinedClass”/>
        <rdfs:subClassOf rdf:resource=”#animal”/>
        <rdfs:subClassOf>
             <oil:NOT>
                  <oil:hasOperand rdf:resource=”#carnivore”/>
             </oil:NOT>
        </rdfs:subClassOf>
    </rdfs:Class>
```

3.5 The DARPA Agent Markup Language (DAML[9])

The *DARPA Agent Markup Language* (DAML) is a US Government-sponsored endeavour aimed to develop a language and tools to facilitate the concept of the Semantic Web. DAML consists of two portions, the ontology language and a language for expressing constraints and adding inference rules. It also includes mappings to other semantic web languages such as SHOE, OIL, KIF, XML, and RDF.

The DAML language is an extension of XML and RDF. The latest release of the language (DAML+OIL) provides a rich set of constructs with which to create machine readable and understandable ontologies and to mark-up information. The ontology language (DAML +OIL) has a well-defined model-theoretic semantics as well as an axiomatic specification that determines the language's intended interpretations. This makes it an unambiguously computer-interpretable language.

The DAML Inference Language is a logical language with a well-defined semantics and the ability to express constraints and rules for reasoning. An example of a constraint is presented in Table 5 using `daml:Restriction` clause.

Table 5. An example of a process[17] description in DAML

```
    <daml:Class rdf:about="Process">
         <rdfs:comment>
              A Process can have at most one name, but names
need not be unique.
         </rdfs:comment>
         <rdfs:subClassOf>
              <daml:Restriction daml:maxCardinality="1">
                   <daml:onProperty rdf:resource="#name"/>
              </daml:Restriction>
         </rdfs:subClassOf>
    </daml:Class>
```

[17] http://www.ai.sri.com/daml/ontologies/services/1-0/Process.daml

DAML+OIL is the result of a merger between DAML and OIL. It is specifically designed for use on the Web. And it exploits XML and RDF, adding the formal rigor of description logic. It takes an object-oriented approach, describing the structure in terms of classes and properties. According to [28], from a formal point of view, DAML+OIL can be seen to be equivalent to a very expressive description logic (DL), with a DAML+OIL ontology corresponding to a DL terminology.

DAML+OIL provides mechanisms for the explicit representation of services, processes, and business models [29].

4 Related Work on the Comparison of Ontology Languages and Ontologies

There are a number of works, where ontology languages and ontologies are compared according to particular criteria. However, each survey is centred on a particular view, like structure, implementation, components, etc., according to which a comparison is made. In this section several works on the comparison of ontology language are presented. For this see Table 6.

As can be seen from Table 6, different criteria are suggested to compare ontology languages. However, the following things are the most important:

- possibility to implement the language or existence of particular tools implementing the selected language and allowing to manage a created ontology;
- optimal number of constructs allowing to define a particular domain of interest;
- possibility to transform a defined ontology into another language.

Table 6. Related work on the comparison of ontology languages

Criteria / Source, Compared languages	Main points of the comparison	Criteria
[2] KIF, F-Logic, Dublin Core, CYC, XML, RDF(S), (KA)^2*, SHOE, OIL, DAML, OWL,	There are not defined concrete criteria. Authors present an overview of ontology languages.	Authors present the following information about languages: Developers; A particular features; Main purpose of a language; Formality; Example; A particular advantages and disadvantages.

Table 6. (*continued*)

[16] CycL, Ontolingua, F-Logic, OCML, LOOM, Telos, RDF(S), OIL, DAML+OIL, XOL, SHOE	Propose quality evaluation framework. Two criteria for ontology language comparison are used: *conceptual basis* of the language and *external representation* of the language. However, they apply only the second criterion in their comparative study.	1. Underlying conceptual basis of the language 2. External representation of the language (domain appropriateness (expressive power and perspectives**), participant knowledge appropriateness, knowledge externalizability appropriateness, comprehensibility appropriateness (number of constructs and abstraction mechanism) and technical actor interpretation appropriateness (formal syntax, formal semantics, inference engine and constraint checking)).
[30] RDF/XML, KIF, Frame-CG and Formalized-English	Shows how RDF/XML, KIF, Frame-CG (FCG) and Formalized-English (FE) can be used for knowledge representation cases. The author highlights various inadequacies, advantages, translations of the selected languages. The article [30] is suggested for knowledge providers as a guide for knowledge representation, and for developers as a list of cases for notations and inferences engines.	1) Conjunctive existentially quantified sentences; 2) Contextualization; 3) Universal quantification; 4) Lambda abstraction, Percentage, Possibility, Valuation; 5) Negations, Exclusions and Alternatives; 6) Collections and Quantifier precedence; 7) Intervals; 8) Function Calls and Lists; 9) Higher-order statements; 10) Declarations and Definitions
[32] WSMO (Web Service Modelling Ontology) and OWL-S	According to authors of [32], WSMO presents some important advantages when compared to OWL-S: a) its conceptual model has a better separation of the requester and provider point of view, b) includes the orchestration of a Web Service enabling its static or dynamic reuse; c) provides formal semantics for the choreography of Web Services and allows multiple ways of interacting with a given service; d) provides a better language layering.	6. Separation of provider and requester point of view 7. Use of non-functional properties 8. Range of non-functional properties 9. Description of requests 10. Mediation 11. Description of orchestration 12. Maturity of the externally visible behaviour specification 13. Formal semantics for the choreography description 14. Multiple choreographies 15. Grounding *16.* Languages

Table 6. (*continued*)

[31] KAoS, Rei and Ponder	The authors suggest adopting ontologies to: a) simplify the task of governing the behaviour of complex environments; b) to simplify description of policies and facilitate the analysis and the careful reasoning over them; c) to simplify the access to policy information	1) *expressiveness* to handle the wide range of policy requirements arising in the system being managed, 2) *simplicity* to ease the policy definition tasks, 3) *enforceability* to ensure a mapping of policy specifications and implementation to various platforms, 4) *scalability* to ensure adequate performance, and 5) *analyzability* to allow reasoning about policies.

* The knowledge annotation initiative of knowledge acquisition

** Perspectives are structural (S), functional (F), behavioural (B), rule (R), Object (O), communication (C), Actor (A).

5 Selecting Criteria for Comparison of Ontology Languages

Based on the related work we use the criteria for the comparison of ontology language as follows:

- ***Constructs or conceptualisation of a domain*** – constructs constitute vocabulary of a domain. Each language uses its own constructs, i.e. types of entities and relations that are considered to exist.
- ***Specification perspective*** – different languages may focus on different perspectives, and may provide constructs for only some perspectives. Authors of [16] define seven specification perspectives: a structural (a static structure); a functional (processes, activities and transformations); behavioural (states and transitions between them); a rule (rules for certain processes, activities, entities); an object (objects, processes and classes); a communication (language actions, meaning and agreement); and actor and role (actors, roles, societies, organizations). A structural perspective is the most important in ontologies.
- ***Expressiveness*** – possibility to express semantics (domain knowledge). I.e. according to the content of a domain knowledge, which can be expressed by a language, it can be lightweight, light heavyweight, and heavyweight (see Section 2). Another important aspect is lexical support – a capability for lexical referencing of elements (e.g., synonyms).
- ***Comprehensibility*** – how easily the language can be understood by the audience [16]. Important aspects are the support of abstraction mechanisms (hiding details), uniform constructs and a reasonable number of phenomena. In [33] authors described four standard hierarchical relations: classification, aggregation, generalization and association. Aggregation is not well supported in many ontology languages [16].
- ***Formality*** – a formal language is made of three components: the syntax (e.g. rules for determining the grammatical well-formedness of sentences), the semantics (e.g. rules for interpreting sentences in a precise, meaningful way within the domain

considered), and the pragmatic (e.g. rules for explaining how to use the language and for inferring useful information from the specification). Otherwise a language is informal.

- ***Inference engine*** – possibility to inference new knowledge from the existing, i.e. possibility of reasoning.
- ***Constraint checking*** – existence of a constraint checking mechanism.
- ***Implementation*** – is a language implemented into a particular tool or not?
- ***Mapping*** with other languages. Nowadays, the feature of a language to map with other language is significant, because of reusability of knowledge and interoperability.

a – Each subsequent sublanguage has its own additional features (http://www.w3.org/TR/owl-features/)

b – Properties can be used to state relationships between individuals or from individuals to data values. Examples of properties include hasChild, hasRelative, hasSibling, and hasAge. Property hierarchies may be created by making one or more statements that a property is a subproperty of one or more other properties (see rdfs:subPropertyOf:).

c – KIF accommodates varying capabilities and/or computational constraints while providing a migration path from more restrictive to more expressive.

d – SWRL: A Semantic Web Rule Language combining OWL and RuleML http://www.daml.org/2004/04/swrl/

e – classification (CL), aggregation (AG), generalization (G), association (AS)

1. ***Paradigm used*** – as presented in Fig. 2, languages can be frame-based, based on first-order predicate logic, description logic and other languages.
2. ***Web standard used*** – some ontology languages are created on the basis of Web standards. Therefore, ontology languages can be initial or based on e.g. derived.
3. ***Standard*** – is the language accepted as a standard?
4. ***Popularity*** – we add this criterion to determine the popularity of a language. We determine it according to the number of links in Google Scholar.

As can be seen, thirteen criteria are selected to compare ontology languages. However, it is important to note that some of those criteria are dependent from other. In this paper we are not presenting a deep analysis of interdependence of criteria. However, it is a topic for future research.

1. Only formal language can be implemented. However, not all formal languages are implemented till now.
2. Formality of a language influences its expressiveness. As a rule, the more a language is formal, the less it is expressive. For example, not all formal languages allow defining of synonym or aggregation relationships.
3. Only formal languages have inference engine. In non-formal or partially formal languages explicit inference can't be warranted. For example, in OWL Full inference is not clear in partial cases. As said in [34]: "It is unlikely that any reasoning software will be able to support complete reasoning for every feature of OWL Full."

Table 7 presents the comparison of four ontology languages according to the selected criteria. Though, the paper presents DAML and OIL, we think that it is reasonable to analyse features of DAML+OIL, since it joins both features of DAML and OIL. Table 8 presents tools implementing analysed languages.

Table 7. Comparison of ontology languages

Ontology language / Criteria		KIF	OWL Lite	OWL DL	OWL Full	RDF + RDF(S)	DAML + OIL
1. *Constructs or conceptualisation of a domain*		The three layered syntax: *basic characters* (BC); *lexemes* (L) combined from BC; and *expressions* combined from the L.	Main features[a]: a *class* defines a group of individuals with the same *properties*[b]; *individuals*, instances of classes.			Triplet statements *subject – predicate – object*	Structure described in terms of *classes* and *properties*.
2. *Specification perspective*		Structural	Structural			Structural	Structural
3. *Expressiveness*		High	Weak	Medium	High	Medium	*Kinds of axiom* and *Kinds of class constructor*
4. *Comprehensibility*[e]		CL[e]: class, member, etc.; AG: composition(); G: SubClassOf(), SubSet(); AS: see KIF	CL[e]: class, subject area; AG: OWLimports: OWLOntology [0..*]; G: SubClassOf; AS: OWLhasValue, etc.			CL[e]: rdf:type, rdfs:Class; G: rdfs:subClassOf; AS: see RDF	CL[e]: rdf:type, rdfs:Class; G: rdfs:subClassOf; AS: see RDF
5. *Formality*	*Syntax*	Formal	Formal		Formal	Formal	Formal
	Semantics	Formal	Formal		not in all cases	Formal	Formal
6. *Inference engine*		yes	yes	yes	Inference problems [35]	lack of affection inference mechanism	yes
7. *Constraint checking*		weak [c]	good	good	good	weak	supports SWRL[d]
8. *Mapping* with other languages		designed for use in the interchange of knowledge among disparate computer systems	RDF		RDF	OWL, DAML+OIL	RDF
9. *Paradigm used*		First-order predicate logic	DL		Cannot be translated to DL	Object-oriented structure	DL
10. *Web standard used*		no	RDF			XML	XML and RDF
11. *Standard*		yes	yes			yes	no
12. *Popularity*		29 100	429 000			137 000	10 500

Table 8. Implementation of ontology languages

Ontology language	Tool
KIF	*EPILOG* – a common lisp inference system. Epilog 2 is not available for public download. A Web Site: <http://www.cs.rochester.edu/research/epilog/>. *JKP* – a Java KIF Parser which can parse ASCII strings representing sentences in a subset of KIF into a Java representation. It is free source. A Web Site: <http://www.cs.umbc.edu/csee/research/kif/jkp/>. *Protégé*[1] – a variant of KIF, which is basis of Protégé Axiom Language (PAL), used to describe Protégé constraints. It is free source.
OWL	*OWL Validator* – a tool to check OWL mark-up for problems beyond simple syntax errors. It accepts ontologies written in RDF/XML, OWL/XML, OWL Functional Syntax, Manchester OWL Syntax, OBO Syntax, or KRSS Syntax. A Web Site: <http://www.w3.org/2001/sw/wiki/OWL_Validator>. *Protégé*[1]
RDF + RDF(S)	*Jena Toolkit* – a Java framework for building Semantic Web applications. It provides a programmatic environment for RDF, RDFS and OWL, SPARQL and includes a rule-based inference engine. A Web Site: <http://jena.sourceforge.net/>. *Sesame* – an open-source Java framework for storage and querying of RDF data. A Web Site: <http://www.openrdf.org/>. *KAON* – an open-source ontology management infrastructure targeted for business applications. A Web Site: <http://kaon.semanticweb.org/>. *TRIPLE* – an RDF query, inference, and transformation language for the Semantic Web. TRIPLE allows the semantics of languages on top of RDF to be defined with rules. A Web Site: <http://triple.semanticweb.org/>. *OntoEdit* – a tool which enables creating, browsing, and modifying ontologies. The Ontology Engineer is supported to transform the conceptual representation to all major ontology representation languages, like RDF(S), XML, DAML+OIL or F-Logic. It is a commercial tool, but a demo version, supporting up to 50 concepts, is available. A Web Site: <http://www.ontoprise.de/>. *Protégé*[1] – an ontology editor and a knowledge-base editor.
DAML + OIL	*OilEd* – an ontology editor allowing the user to build ontologies using DAML+OIL, it includes a FaCT (Fast Classification of Terminologies) reasoner for inferencing. *DAML Viewer* – a small Java tool for quick navigation through a DAML+OIL ontologies. A Web Site: <http://www.daml.org/viewer/>. *DAMLJessKB* – a library which uses RDF API to read in DAML files and feeds the triples into JESS, a Java-based expert system. The rules in the expert system apply the DAML semantics to the triples, creating new facts. A Web Site: <http://edge.cs.drexel.edu/assemblies/software/damljesskb/>. *SweetJess* – Semantic Web Enabling Technlogies for Jess. It Converts RULEML into DAML RuleML and then converts this into JESS Rules. A Web Site: <http://userpages.umbc.edu/~mgandh1/>.

Many of RDF tools, like Jena, Sesame, OntoEdit, support DAML+OIL.

Protégé-Frames ontologies can be exported into a variety of formats including RDF(S), OWL, and XML Schema.

6 Discussion

As the previous sections made clear, each ontology language exhibits advantages and disadvantages and thus the choice of an ontology language should be driven by the purpose what for it will be used. Of course, as can be seen from the research, OWL claim to be the most popular, having the biggest community and the most spreading and applicable semantic Web language. However, the main drawback of OWL is lack of supporting tools. At this moment we found only two tools – *Protégé* and *OWL Validator* – supporting OWL and allowing to develop OWL ontologies. While, KIF and RDF + RDF(S) having disadvantages, like heavyweight in the first case and lack of affection inference mechanism and the lack of formal underlying model semantics in the second case, they have the broader application. I.e. KIF is used as a basis in SUMO (SUO-KIF language), Protégé PAL. RDF is used as a basis in OWL, DAML, Protégé and FOAF[18].

DBpedia[4] uses the RDF as a flexible data model for representing extracted information and for publishing it on the Web. SPARQL[19] – a query language for RDF – used to query DBpedia data.

Recently, WordNet[7] [36] has been adopted in the Semantic Web research community. It is used mainly for *annotation* and *retrieval* in different domains, like cultural heritage [37] and product catalogs [38], to *ground other vocabularies*, like FOAF, and as *background knowledge* in ontology alignment tools and other applications[20]. Therefore, a number of conversions of WordNet (e.g. WordNet's Prolog format) in RDF and/or OWL [39], [40], [41] are created.

Another similarity used in all languages is a *layered approach* and the use of *standards*. A number of languages have been developed on the basis of a particular standard. For example, RDF is used for developing OWL, XML is used for developing RDF, etc. The main advantage of using a layered approach for creating a particular language is improving and refining existing language. For example, RDF is improved and extended to create OWL.

Recently, a *sublanguage* or *module approach* is also popular. It means that language consists of several modules or sublanguages, where each sublanguage complements the previous sublanguage. This statement can be defined by equation (1).

$$L = \{L_i, i = 1, ..., n, L_i < L_{i+1}\}, \tag{1}$$

where L is a language consisting of sublanguages L_i and each subsequent sublanguage complements a preceding sublanguage.

This approach is used in OWL, which consists of OWL Lite, OWL DL and OWL Full, and DAML+OIL. Another example is SWRL, allowing users to write rules in terms of OWL concepts. It is created to provide more powerful deductive reasoning capabilities than OWL alone. Semantically, SWRL is built on the same description logic foundation as OWL and provides similar strong formal guarantees when performing inference.

---

[18] http://xmlns.com/foaf/spec/

[19] http://www.w3.org/TR/rdf-sparql-query

[20] http://en.wikipedia.org/wiki/WordNet

The main advantage of the sublanguage approach is that an existing language can be extended by creating a sublanguage without necessity to define a new language with a set of new constructs.

Coding examples from the literature have been presented for all languages. The presented research can be refined and extended according to new ontology languages and criteria of comparison.

References

1. Berners-Lee, T.: The Semantic Web as a language of logic (2009), `http://www.w3.org/DesignIssues/Logic.html`
2. Gutierrez-Pulido, J.R., Ruiz, M.A.G., Herrera, R., Cabello, E., Legrand, S., Elliman, D.: ntology languages for the semantic web: A never completely updated review. Knowl.-Based Syst., 489–497 (2006)
3. Guarino, N.: Formal Ontology and Information Systems. In: Proc. of FOIS 1998, Trento, Italy, pp. 3–15. IOS Press, Amsterdam (1998)
4. OntologyDefinition Metamodel. Object Management Group, OMG (2009), `http://www.omg.org/spec/ODM/`
5. Kalibatiene, D., Vasilecas, O.: Ontology Axioms for the Implementation of Business Rules. Technological and Economic Development of Economy 16(3), 471–486 (2010)
6. Vasilecas, O., Kalibatiene, D., Guizzardi, G.: Towards a Formal Method for Transforming Ontology Axioms to Application Domain Rules. Information Technology and Control 38(4), 271–282 (2009)
7. de Almeida Falbo, R., Silva de Menezes, C., da Rocha, A.R.C.: A Systematic Approach for Building Ontologies. In: Coelho, H. (ed.) IBERAMIA 1998. LNCS (LNAI), vol. 1484, pp. 349–360. Springer, Heidelberg (1998)
8. Culmone, R., Rossi, G., Merelli, E.: An Ontology Similarity algorithm for BioAgent. In: Ercolani, S., Zamboni, M.A. (eds.) NETTAB Workshop on Agents and Bioinformtics, Bologna (2002), `http://www.bioagent.net/WWWPublications/Download/NETTAB02P1.pdf`
9. Lin, S., Miller, L.L., Tsai, H., Xu, J.: Integrating a Heterogeneous Distributed Data Environment with a Database Specific Ontology. In: Sha, E.H.M. (ed.) ISCA PDCS, pp. 430–435 (2001)
10. McGuinness, D.L., Patel-Schneider, P.F.: Usability Issues in Knowledge Representation Systems. In: Mostow, J., Rich, C. (eds.) AAAI/IAAI, pp. 608–614 (1998)
11. Noy, N.F., Fergerson, R.W., Musen, M.A.: The Knowledge Model of Protégé 2000: Combining Interoperability and Flexibility. In: Dieng, R., Corby, O. (eds.) EKAW 2000. LNCS (LNAI), vol. 1937, pp. 17–32. Springer, Heidelberg (2000)
12. Gruber, T.R.: A Translation Approach to Portable Ontology Specifications. Knowledge Acquisition 5, 199–220 (1993)
13. Ontology language. Wikipedia (2010), `http://en.wikipedia.org/wiki/Ontology_language`
14. Lavbic, D., Krisper, M.: Facilitating Ontology Development with Continuous Evaluation. Informatica, Lith. Acad. Sci., 533–552 (2010)
15. Labic, D., Krisper, M.: Rapid Ontology Development. In: EJC, pp. 283–290 (2009)
16. Su, X., Ilebrekke, L.: A Comparative Study of Ontology Languages and Tools. In: Pidduck, A.B., Mylopoulos, J., Woo, C.C., Ozsu, M.T. (eds.) CAiSE 2002. LNCS, vol. 2348, pp. 761–765. Springer, Heidelberg (2002)

17. Genesereth, M.R.: Knowledge Interchange Format (KIF). Draft proposed American National Standard (dpANS), NCITS.T2/98-004. Stanford University (2004), `http://logic.stanford.edu/kif/dpans.html`
18. Patil, R.S., Fikes, R.E., Patel-Schneider, P.F., Mckay, D., Finin, T., Gruber, T., Neches, R.: An Overview of the DARPA Knowledge Sharing Effort. In: Huhns, M.N., Singh, M.P. (eds.) KR 1992, pp. 243–254. Morgan Kaufmann Publishers Inc., San Francisco (1992)
19. Berendt, B., Hotho, A., Mladenič, D., van Someren, M., Spiliopoulou, M., Stumme, G.: A Roadmap for Web Mining: From Web to Semantic Web. In: Berendt, B., Hotho, A., Mladenič, D., van Someren, M., Spiliopoulou, M., Stumme, G. (eds.) EWMF 2003. LNCS (LNAI), vol. 3209, pp. 1–22. Springer, Heidelberg (2004)
20. Antoniou, G., Harmelen, F.V.: Web Ontology Language: OWL. Handbook on Ontologies, 67–92 (2004)
21. Drummond, N., Horridge, M., Stevens, R., Wroe, C., Sampaio, S.: Pizza ontology. The University of Manchester (2007), `http://www.co-ode.org/ontologies/pizza/2007/02/12/pizza.owl`
22. Schwitter, R.: A Controlled Natural Language Layer for the Semantic Web. In: Zhang, S., Jarvis, R.A. (eds.) AI 2005. LNCS (LNAI), vol. 3809, pp. 425–434. Springer, Heidelberg (2005)
23. Kothari, C.R., Russomanno, D.J.: Relation Elements for the Semantic Web. In: Wolff, K.E., Pfeiffer, H.D., Delugach, H.S. (eds.) ICCS 2004. LNCS (LNAI), vol. 3127, pp. 275–286. Springer, Heidelberg (2004)
24. Broekstra, J., Klein, M.C.A., Decker, S., Fensel, D., Harmelen, F.V., Horrocks, I.: Enabling knowledge representation on the Web by extending RDF Schema. Computer Networks, 609–634 (2002)
25. Fensel, D., Harmelen, F.V., Horrocks, I., McGuinness, D.L., Patel-Schneider, P.F.: OIL: An Ontology Infrastructure for the Semantic Web. IEEE Intelligent Systems, 38–45 (2001)
26. Jong-Soo, S.: Ontology Languages. Korea University (2010), `http://onto.tistory.com/attachment/4974544093158B0.ppt`
27. Broekstra, J., Klein, M., Decker, S., Fensel, D., Horrocks, I.: Adding formal semantics to the Web building on top of RDF Schema. In: ECDL 2000 Workshop on the Semantic Web (2000), `http://www.cs.vu.nl/~mcaklein/presentations/2001-05-02-WWW10-HongKong.pdf`
28. Horrocks, I.: DAML+OIL: a Description Logic for the Semantic Web. IEEE Data Eng. Bull., 4–9 (2002)
29. Hendler, J.A.: Agents and the Semantic Web. IEEE Intelligent Systems, 30–37 (2001)
30. Martin, P.: Knowledge representation in CGLF, CGIF, KIF, frame-CG and formalized-english. In: Priss, U., Corbett, D.R., Angelova, G. (eds.) ICCS 2002. LNCS (LNAI), vol. 2393, pp. 77–91. Springer, Heidelberg (2002)
31. Tonti, G., Bradshaw, J.M., Jeffers, R., Montanari, R., Suri, N., Uszok, A.: Semantic Web Languages for Policy Representation and Reasoning: A Comparison of KAoS, Rei, and Ponder. In: Fensel, D., Sycara, K., Mylopoulos, J. (eds.) ISWC 2003. LNCS, vol. 2870, pp. 419–437. Springer, Heidelberg (2003)
32. Lara, R., Roman, D., Polleres, A., Fensel, D.: A Conceptual Comparison of WSMO and OWL-S. In: Zhang, L.-J., Jeckle, M. (eds.) ECOWS 2004. LNCS, vol. 3250, pp. 254–269. Springer, Heidelberg (2004)
33. Krogstie, J., Soelvberg, A.: Information systems engineering – concepttual modeling in a quality perspective. Norwegian University of Technology and Science (1999)

34. McGuinness, D.L., van Harmelen, F. (eds.): OWL Web Ontology Language Overview. W3C Recommendation (2004), http://www.w3.org/TR/owl-features/
35. Horrocks, I.: OWL: A Description Logic Based Ontology Language. In: Gabbrielli, M., Gupta, G. (eds.) ICLP 2005. LNCS, vol. 3668, pp. 1–4. Springer, Heidelberg (2005)
36. Fellbaum, C. (ed.): WordNet: An Electronic Lexical Database. Bradford Books (1998)
37. Hollink, L., Schreiber, A.T., Wielemaker, J., Wielinga, B.J.: Semantic annotation of image collections. In: Handschuh, S., et al. (eds.) KCAP 2003, pp. 41–48 (2003)
38. Guarino, N., Masolo, C., Vetere, G.: Ontoseek: Content-based access to the web. IEEE Intelligent Systems 14(3), 70–80 (1999)
39. van Assem, M., Gangemi, A., Schreiber, G.: Conversion of WordNet to a standard RDF/OWL representation. In: LREC 2006 (2006)
40. van Assem, M., Gangemi, A., Schreiber, G. (eds.): RDF/OWL Representation of WordNet. W3C Working Draft (2006), http://www.w3.org/TR/wordnet-rdf/
41. WordNet in RDF/OWL, Best Practice (2006), http://ebiquity.umbc.edu/blogger/2006/02/20/wordnet-in-rdfowl-best-practice/

Advanced RDB-to-RDF/OWL Mapping Facilities in RDB2OWL

Guntars Būmans[1,*] and Kārlis Čerāns[2,**]

[1] Department of Computing, University of Latvia, Raiņa bulvāris 19, Riga LV-1586, Latvia
guntars.bumans@gmail.com
[2] Institute of Mathematics and Computer Science, University of Latvia
Raiņa bulvāris 29, Riga LV-1459, Latvia
karlis.cerans@lumii.lv

Abstract. We present advanced features of RDB2OWL mapping specification language that allows expressing RDB-to-RDF/OWL mappings in a concise and human comprehensible way. The RDB2OWL mappings can be regarded as documentation of the database-to-ontology relation. The RDB2OWL language uses the OWL ontology structure as a backbone for mapping specification by placing the database link information into the annotations for ontology classes and properties. Its features include reuse of database table key information, user defined scalar and aggregate functions, table-based functions and multiclass conceptualization that is essential to keep mappings compact in case when large tables are mapped onto several classes on the table field value existence basis.

Keywords: Relational databases, RDF, OWL ontologies, mappings.

1 Introduction

The use of Semantic Web technologies has grown dramatically in recent years. It has been motivated by the need for semantic information organization both on global and on enterprise scale. Various open standards and notations have been developed such as RDF[1], SPARQL 1.1 [2], OWL 2.0 [3] to serve this purpose. RDF is data formalism specifying information coding in a triples form as a graph. The schema information for the graph may be given as RDF Schema [4] or OWL ontology. Most of the data still reside in relational databases with no near future vision for massive global RDB-to-RDF triple store migration. This calls for development of technologies for relational database integration into new Semantic Web data world. For this purpose development has been directed towards RDB-to-OWL/RDF mapping specification means (languages, techniques) and machine processing of mappings to generate RDF triples from RDB data and/or to represent RDB data as RDF triples.

A number of techniques and tools have been developed for these tasks, including R_2O [5], D2RQ [6], Virtuoso RDF Views [7], DartGrid [8] as well as UltraWrap [9]

* Partially supported by ESF project 2009/0138/1DP/1.1.2.1.2/09/IPIA/VIAA/004.
** Partially supported by ERAF project 2011/0009/2DP/2.1.1.1.0/10/APIA/VIAA/112.

J. Grabis and M. Kirikova (Eds.): BIR 2011, LNBIP 90, pp. 142–157, 2011.

and Triplify [10]. There is upcoming W3C standard R2RML [11] for RDB-to-RDF mapping specification, as well as a standard for direct (technical) mapping [12] of RDBs to RDF format. In the case of the latter approach the obtained data that correspond to the "technical" data schema can be afterwards transformed into a conceptual one either by means of SPARQL Construct queries, as in Relational.OWL [13] approach, or by means of some RDF-to-RDF mapping language such as R2R [14], or some model transformation language (see e.g. [15] for an example approach).

The RDB2OWL approach we are elaborating here deals with defining mappings between source database schema and conceptual data ontology that may substantially differ from the "technical" presentation of the database schema. The existing R_2O, D2RQ, Virtuoso RDF Views and R2RML languages already cover well the low-level constructs that are essential in this context.

Within the RDB2OWL approach we focus on offering a mapping specification language [16, 17] that allows creating human readable mapping specifications and attaching those as annotations to the target ontology classes and properties. The main contribution of this paper is the description of the advanced high-level constructs in the RDB2OWL mapping specification language that allows keeping the mapping definition fully human-comprehensible also in the case of complex mapping structure.

The RDB2OWL execution environment is technically based on a designated relational database schema to store the mapping information, and the triple generation is done by two-phase SQL processing (the first phase processes mapping information to generate SQL sentences for triple creation from source database and the second phase execute the SQL scripts generated in the first phase) [18, 19]. This approach benefits from SQL processing speed of modern RDBMS, therefore, the combination of a high level specification language with efficient implementation structures is achieved. As a future perspective: RDB2OWL human readable mapping language can be transformed to R2RML language to use any tool that implements it.

After presenting a brief review of basic RDB2OWL notations, we move on to explanation of advanced mapping constructions:

- multiclass conceptualization to obtain compact mapping specifications for a parent class and its subclasses mapped to the same database table;
- creation and usage of auxiliary database schema objects that can help defining the source database to ontology mapping;
- user defined and built-in RDB2OWL scalar and aggregate functions; function definition expressions can include references to source and auxiliary database tables and columns to enhance expressiveness.

We discuss both the basic and advanced RDB2OWL mapping specification language on a simple mini-University example [17], as well as Latvian medicine registries RDB-to-RDF/OWL migration case [20, 21].

2 RDB2OWL Core Language: An Overview

A RDB2OWL mapping is a relation from a source RDB schema S to target OWL ontology O, specifying the correspondence between the source database data and RDF triples "conforming" to the target ontology. The RDB schema to OWL ontology mapping consists of individual entity-level mappings, each attached to an OWL entity (class or property) specifying how to generate RDF triples for this entity from relational DB table data. A mapping for an OWL class is called *class map*, mapping for an OWL property - *property map*. There are *datatype property maps* and *object property maps* for OWL datatype and object properties respectively.

We illustrate the basic mapping notions and syntax by mini-University example adopted from [22]. A database schema and corresponding target ontology for the example is shown in Figure 1. The basic mapping building pattern is in marking OWL classes as corresponding to relational tables, data properties to table columns and object properties to foreign key relations. The example shows, however, that even in small-sized cases the mappings are not exactly one-to-one. For example, a class *Course* that corresponds to table *XCourse* is split into subclasses on the basis of *required* column of *XCourse* table (there is a similar split for *Teacher* class). There is a class *Person* that covers *Student* and *Teacher* classes in the ontology. The property *takes* is *Student*-to-*Course* n:n relation, it corresponds to *XStudent-XRegistration-XTeacher* table pattern (the RDB model doesn't support n:n relations between tables).

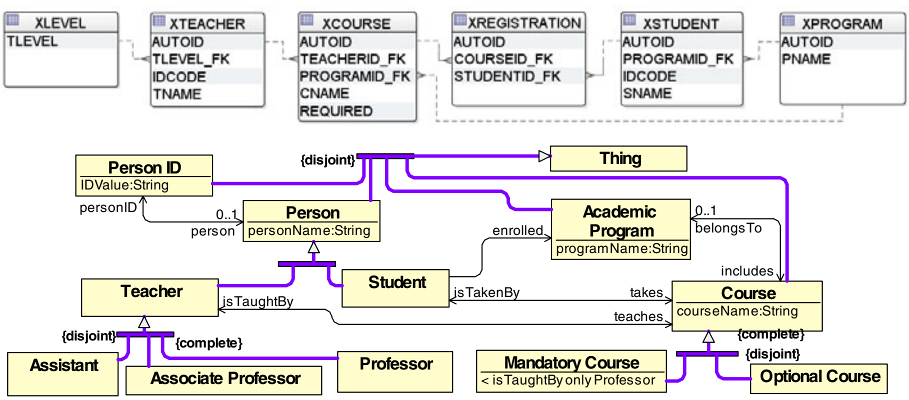


Fig. 1. A RDB schema and ontology of mini-University

Figure 2 shows mini-University ontology annotated in accordance with RDB2OWL core mapping language. A custom extension of OWLGrEd editor [23] is used for showing graphically ontology together with annotations; the *DBExpr*-annotations for classes and properties are depicted as '{DB: <annotation text>}'.

Some simple class map examples in Fig.2 are *XStudent*, *XTeacher*, *XCourse;* each of them relates an ontology class to a database table. The class maps for derived classes (e.g. the *Assistant* class) in the example include a reference to the sole class map defined at another class (e.g., *[[Teacher]]*), followed by a filter (e.g. *Level='Assistant'*). The class *PersonID* intuitively is based on *IDCode* column for both *XTeacher* and *XStudent* tables; formally this dependence is described by ascribing two named class maps (*S* and *T*) to *PersonID* class; each of these class maps specify both the table on which the *PersonID* class is based on and the way of *PersonID* class instance URI formation on the basis of the data contained in *XTeacher* or *XStudent* table row, corresponding to this instance.

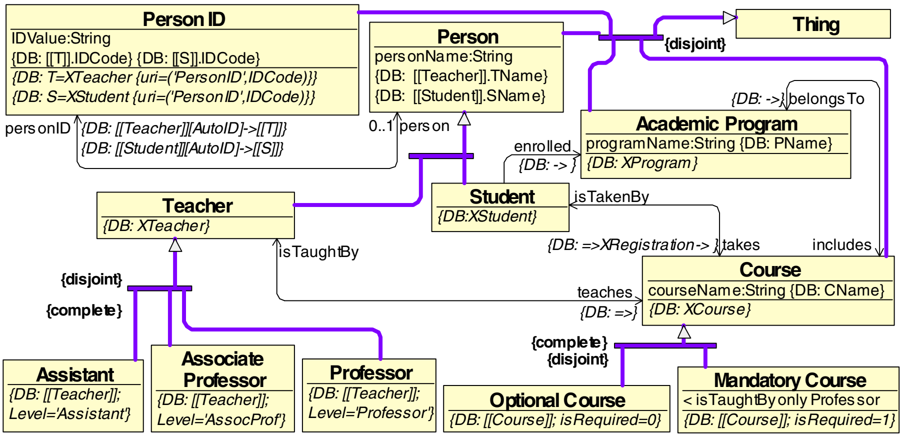


Fig. 2. Annotated mini-University ontology using RDB2OWL Core model [17]

An object property map is defined as a table expression typically involving explicitly or implicitly specified tables corresponding to the property domain and range classes. So, the property map for property *enrolled*, depicted in the diagram as ->, could be described also in a more detailed way as <*s*>-><*t*> (<*s*> and <*t*> are "reference marks" for sole class maps ascribed to the domain and range classes of the property), or as *(XStudent <s>)->(XProgram <t>)* (the referred table names made explicit), or *(XStudent <s>)[ProgramID_FK]->[AutoID](XProgram <t>)* (explicit specification of table linking columns), or *(XStudent <s>), (XProgram <t>); <s>.ProgramID_FK = <t>.AutoID* (a simple table expression form without navigation expressions). The expanded expression forms can be used instead of the simple form whenever this is convenient. Omitting the column information within the navigation link is possible whenever the navigation link corresponds to the sole foreign key to primary key relation between two tables (for the relation in a "backwards direction" the column information may be omitted, as well, while changing the navigation link symbol from -> to =>).

The navigation expressions can be chained, as in the object property map for *takes* property: *=>XRegistration->*. By inserting the implicit tables, it is expanded to *(XStudent <s>)=>XRegistration->(XCourse <t>)*.

Syntactically a table expression consists of a comma-separated list of *reference items*, optionally after semicolon followed by a filter expression. Each reference item can be a table name or a nested table expression, followed optionally by an alias. Some table expression examples are, as follows:

1. *XStudent*
2. *XRegistration R, XCourse C, XTeacher T;*
   R.CourseID_FK=C.AutoID AND C.TeacherID_FK=T.AutoID
3. *(XStudent S, XProgram P; X.ProgramID_FK=P.AutoID) SP,*
   (XRegistration R, XCourse C; R.CourseID_FK=C.AutoID) RC;
   SP.(S.AutoID)= RC.(R.StudentID_FK)

Furthermore, a reference item can be expressed as a *navigation list* e.g., *XStudent [ProgramID_FK]->[AutoID]XProgram* where *[ProgramID_FK]->[AutoID]* is a *navigation link* which can be simplified to ->, as described above. Navigation links can be based also on comparison of *value expression* evaluation in respective tables, such as *(Person P1)[salary]->[salary*2] (Person P2)*, or on column list correspondence, as in: *(XRegistration R1)[CourseID_FK, StudentID_FK]-> [CourseID_FK, StudentID_FK] (XRegistration R2).*

The navigation items in table expressions may include also row filtering conditions: *XStudent:(sname='Alice')[ProgramID_FK]->[AutoID] Xprogram.*

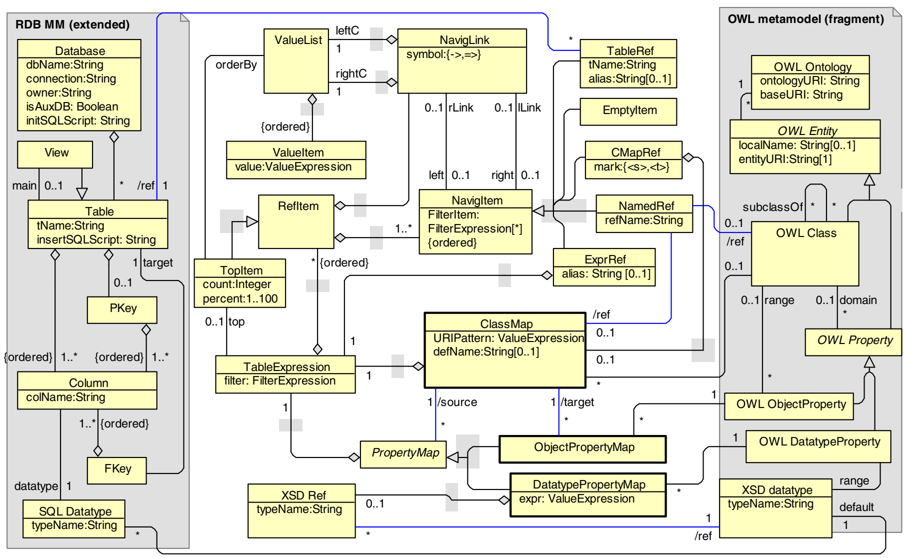


Fig. 3. RDB2OWL Core metamodel

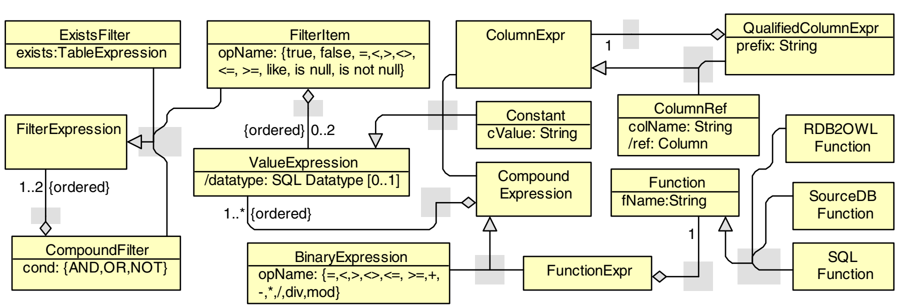


Fig. 4. Expression and filter metamodel

A *datatype property* map is described by a table expression that is required to contain (explicitly or implicitly) a single <s>-marked reference to its source class map (typically, the sole class map for the property domain class); followed by a value expression after a dot, and further on by an optional datatype specification preceded by '^^'. The following are examples for property map specification:

- *XCourse <s>->XProgram P.concat(<s>.cname, ", ", P.pname)*
- *XCourse <s>.cname*
- *<s>.cname*
- *cname* ('<s>.' can be omitted)

Figure 3 summarizes the basic notions of RDB2OWL Core mapping language in a metamodel form; Figure 4 provides a supplementary expression and filter metamodel.
A more detailed description of core RDB2OWL mapping constructs is in [17].

3 Advanced RDB2OWL Constructs

RDB2OWL core mapping language can be sufficient for simple applications but real life practical use cases show the need for various extensions while keeping the mappings compact and intuitive. The extensions described in this section are related to their practical use in a case study of using RDB2OWL approach to migration into RDF format of 6 Latvian medical registries [20, 21]. In the considered example the relational database contained 3 GB data consisting of 106 tables, 1353 table columns and 3 million rows in total, as reported in [18, 19]. The corresponding ontology had 172 OWL classes, 814 OWL datatype properties and 218 OWL object properties [19].

3.1 Multiclass Conceptualization

The meta-models of OWL ontology and RDB schema differ in that the former foresees a subclass relation, while the latter does not. We enhance RDB2OWL mapping language to deal with use cases where this difference is exploited. A *multiclass conceptualization* is a mapping pattern where one database table T is mapped to several ontology classes C1, C2,…, Cn each one reflecting some subset of T columns as the class' properties. In a standard way one would map each of Ci to the table T and would add to the respective class maps for Ci filtering expressions stating that only those rows of T correspond to Ci instances where at least one of the columns from the column set corresponding to the class' property column set has been filled. Mappings of Latvian Medicine registries contain such patterns where tables with several hundred columns are split into subsets of 20-30 columns. Filtering conditions for these mappings are lengthy and difficult to write and read.

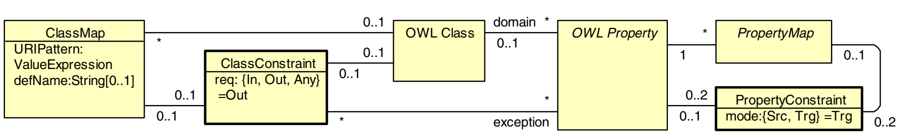


Fig. 5. Class and property constraint metamodel

To handle issue we introduce a *ClassConstraint* class whose instances specify requirement: a RDF triple <*x,’rdf:type’,o*> can exist in the target triple set only if a triple <*x,p,y*> exists for some property *p* with domain *o* and some resource *y* (in the terms of the last paragraph x would be an individual corresponding to a row in table T and o would be some class Ci).

If a class constraint is attached to an OWL class *c* it means that all generated instances *x* of *c* (the triples <*x,’rdf:type’,c*>) should be checked for existence of property *p* instances for incoming (*p.range=o*), outgoing (*p.domain=o*, the default) or any (incoming or outgoing) properties. If a class constraint is attached to a class map, then it applies only to class instances that are created in accordance to this class map. The *exception* link from a class constraint specifies what properties are not to be looked at when determining the property existence. In Latvian Medical registries we have used class level constraints for 54 out of 172 OWL classes with 514 out of 814 OWL datatype properties belonging to these constrained classes.

A *PropertyConstraint* class instance attached to a property *p* means requirement: check if for the subject *s* (*mode=Src*) or object *t* (*mode=Trg*) from a <*s,p,t*> triple there exists the triple <*s,*’rdf:type’*, p.domain* > (or <*t,*’rdf:type’*, p.range*>) generated by the mapping (if the check fails, delete the <*s,p,t*> triple). The checks associated with property constraints are to be applied after the class constraint resolution. Note that the property constraints with *mode=Trg* apply only for object properties. In Latvian Medicine registries there is a case of sugar diabetes mapping where property constraint appear essential in conjunction with class constraint use.

The class and property constraints are part of the mapping definition, not part of the target OWL ontology. The meaning of these constraints is fully “closed world”: delete the triple, if the additional context is not created by the mapping.

3.2 Auxiliary Database Objects

There are cases when direct mapping between source RDB and the target ontology is not possible or requires complex expressions involving manual SQL scripts. Additional databases and its tables can be introduced for the mapping purpose. There can be multiple *Database* class instances in RDB2OWL core metamodel (see Figure 3). If the value of *isAuxDB* attribute is *true* then the database is auxiliary; otherwise it is a source database. A SQL script can be executed (attribute *initSQLScript*) to create necessary schema objects in auxiliary database and populate the tables with the needed data (attribute *insertSQLScript* of *Table* class). The definition and data of new auxiliary schema objects are considered to be part of the mapping specification.

The auxiliary tables and views can be used to simplify mapping presentation.

Another, more fundamental, usage context for auxiliary tables is ontology class or property that would naturally correspond to a database schema object that does not exist in the source RDB schema. A typical case of this category is a non-existing classifier table, which naturally appears in the ontological (conceptual) design of the data. In Fig.6, the OWL class *PrescribedTreatment* is based on database table *PatientData*. The *PatientData* table has “similar” binary attributes indicating that certain treatments on the patient have been performed. In the ontological modeling one would introduce a single *diabetesTreatment* property to reflect all the “similar” fields from the *PatientData* table, the different fields being distinguished by different instances of the *DiabetesTreatment* class. The instances within the *DiabetesTreament* class may be specified either by directly entering them into the target ontology, or one could create an extra classifier table within an auxiliary database (a *TreatmentCategory* table in the example) that can be seen as a source for *DiabetesTreatment* instances.

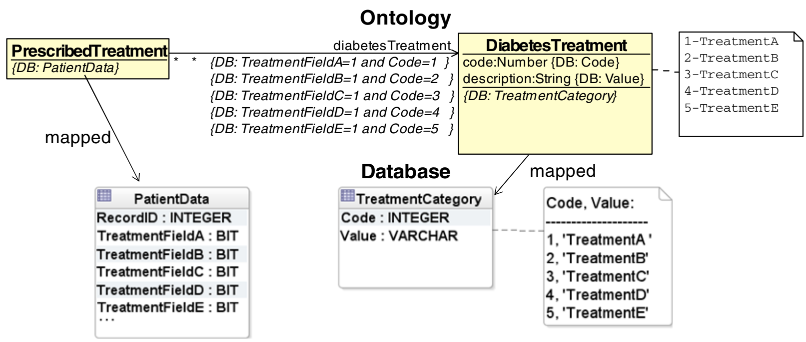


Fig. 6. Ontology and Database fragment for Diabetes Treatment modeling

3.3 RDB2OWL Functions

Possibility of function definition and use increases substantially the abstraction level of programming notation. In practical RDB2OWL mapping use cases the functions have been important e.g. to cope concisely with legacy design patterns present in the source database. A basic RDB2OWL function metamodel is shown in Figure 7.

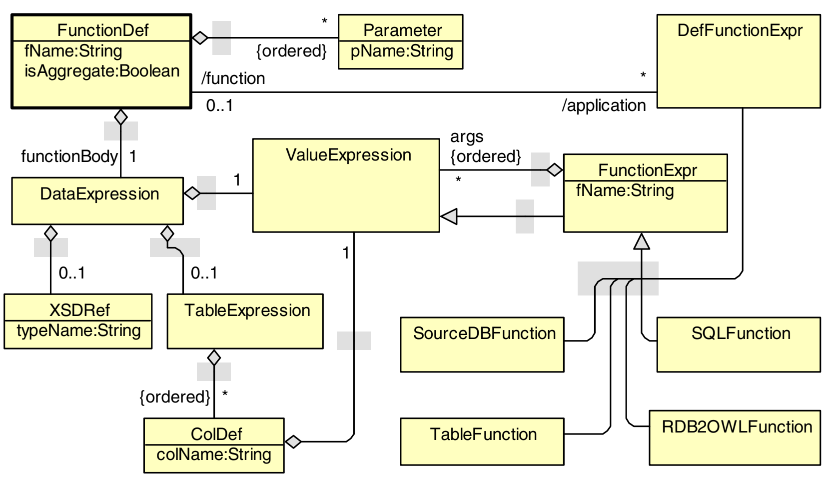


Fig. 7. RDB2OWL Function metamodel

We introduce scalar-argument as well as aggregate functions into RDB2OWL (aggregate functions are shown in Figure 8 and are described in section 3.4). The scalar-argument (non-aggregate) functions in RDB2OWL are:

1. Built-in functions (class *RDB2OWLFunction*),
2. User defined functions (class *FunctionDef* for definition and associated class *DefFunctionExpr* for application),
3. Functions based on stored functions in the source database (class *SourceDBFunction*),
4. Functions whose argument-value pairs are stored in table with two columns (class *TableFunction*) and
5. SQL functions (class *SQLFunction*).

3.3.1 Built-in Functions

There are some functions that are frequently needed in different concrete mapping cases. For example, SQL numeric literals 1 / 0 generally are used for boolean *true* / *false* values therefore we have rationale to build-in *iif* function. For every mapping case the ultimate target is generated triples set, the function *uri* may be helpful for custom URI pattern definition. Built-in function names are prefixed by # to distinguish from user-defined functions. Function parameter names are prefixed by @.

Table 1. Built-in RDB2OWL functions

#varchar(@a)	Converts a single argument to SQL varchar type
#xvarchar(@a)	Converts a single argument to varchar, eliminates leading and trailing spaces
#concat(…)	Takes any number of arguments, converts them into the SQL varchar type and then concatenates
#xconcat(…)	Takes any number of arguments, converts into the SQL varchar type, eliminates leading and trailing spaces and then concatenates
#uri(@a)	Converts a single argument to varchar, eliminates leading and trailing spaces, converts to uri encoding
#uriConcat(...)	Takes any number of arguments, converts them into the SQL varchar type, eliminates leading and trailing spaces, converts to uri encoding and then concatenates
#exists(...)	Can take any number of arguments and returns 1, if at least one argument is not null, otherwise returns 0. The form *#exists(Col1, Col2,..,Colk)* is used in Latvian Medicine registry case as an alternative to multiclass conceptualization approach, if k is small.
#iif(@a,@b,@c)	Chooses the value of b or c depending on a value being 1 or 0. Example: *#iif(is_resident,'true','false')*
#all(...)	Can take any number of arguments and returns 1, if all arguments are not null, otherwise returns 0.

The list of built-in functions is to be finalized within upcoming RDB2OWL reference manual.

3.3.2 User Defined Functions

An important feature of RDB2OWL is possibility for user-defined functions which can be referenced from class map and property map definitions. Function value is obtained by evaluating its value expression in the context of the function call. The definition of a simple function (e.g., f(@x)=2*@x+1) consists just of value expression, referring to function parameters. For simple functions no *TableExpres-sion* instance is linked to *DataExpression* instance (see metamodel in Figure 7).

A user-defined function, however, may include also a table expression (an additional data context for expression evaluation) and a list of column expressions (=calculated columns) relying both on function's arguments and function's table expression and used in further value expression evaluation. Syntactically we have the function definition in the following form:

$f(@X_1, @X_2, \ldots, @X_n)=(T; filter; colDef_1, \ldots, colDef_m).val$ *^^xsd_datatype*

where *T;filter* is a table expression and each $colDef_i$ is in form $var_i=e_i$ for a value expression e_i. The table expression with column definitions, as well as optional datatype specification (*^^xsd_datayppe*) may be omitted. When the defined function is called as $f(V_1, V_2, \ldots, V_n)$ in some table context *A*, it is evaluated as:

$(A,T;filter'; colDef'_1, \ldots, colDef'_m).val[V_1/@X_1, \ldots, V_n/@X_n]$, where $[V_1/@X_1, \ldots, V_n/@X_n]$ means substitution of the value V_i for the variable $@X_i$ for all i,
$filter' = filter[V_1/@X_1, \ldots, V_n/@X_n]$ and each $colDef'_i = colDef_i [V_1/@X_1, \ldots, V_n/@X_n]$.

As simple function example with no tables attached is function that converts integer values of 0 / 1 to *'true^^xsd:Boolean' /'false ^^xsd:Boolean'* is:
BoolT(@X) = #iif (@X,'true','false')^^xsd:Boolean.

Another simple example: *Plus(@X, @Y) = @X + @Y.*

In Latvian medical registries there have been numerous situations where many year values were stored in one varchar type field value (e.g., '199920012005') but corresponding datatype property having separate instances for each value {'1999', '2001', '2005'}. The value splitting can be implemented by joining the source table with auxiliary table *Numbers* having single integer type column filled with values from 1 to 999 (see [24]), as in the function:

*split4(@X)=((Numbers;len(@X)>=N*4).substring(@X,N*4-3,4)),* The application *split4(FieldX)* then splits character string into set of substrings of length 4.

If calculated values $colDef_i$: $var_i=e_i$ are included in the function definition, these can be referenced from the function's value expression. This enables to write more structured and readable code. A simple example function that takes values from two tables and stores intermediate values in variables *courseName* and *teacherName* is:

```
FullCourseInfo(@cId)=((XCourse c)->(XTeacher t);
c.AutoId=@cId;
courseName=#concat(c.CName, #iif(c.required,'
required',' free')),
teacherName=t.TName)
.#concat(courseName, ' by ', teacherName)
```

We can look on the database table with two columns $T(C_1, C_2)$ as a storage structure with rows containing argument-value pairs of some function *f*. We call this function *f* a *table function* based on table *T*. If the column C_1 has a unique constraint (e.g., a primary key column), *f* is a single-valued function; otherwise *f* is multi-valued function. Multi-valued functions are appropriate for property maps of properties with cardinality larger than 1.

A table function based on table $T(C_1, C_2)$ actually is shorthand of user-defined function with table expression comprising table *T*: $f(@X) = (T; C_1=@X).C_2$.

A typical usage of table functions is for classifier tables containing code and value columns. For example, to associate country codes with full country names a table *Country(code varchar(2), description varchar(40))* with data {('de','Germany'), ('en','England'), ('lv','Latvia'),…} could be used for a table function.

3.4 Aggregate Functions

RDB2OWL has aggregate functions (built-in and user defined) whose application to appropriate arguments yield aggregate expressions (*AggregateExpr* instances). Aggregate expression is kind of value expression therefore it can be substituted for value part of datatype property: *T.<aggregate expression>*, where table expression *T* is context in which aggregate function is calculated; we call it a *base table expression* of aggregate function application. An aggregate expression specifies 2 things: which aggregate function *f* to execute (built in S*um, Count, Min, Max, Avg* or user defined) and what data should be passed to *f* in terms of data expression *D* (*DataExpression* instance) that contain an optional table expression *E* (*TableExpression* instance) and a value expression *V* (*ValueExpression* instance).

With these denotations an aggregate expression application takes a form: *T.f(E.V)*, where base table expression *T* is explicitly or implicitly referenced by <*b*>-mark from within *E* table expression reference structure. A base table expression *T* can be

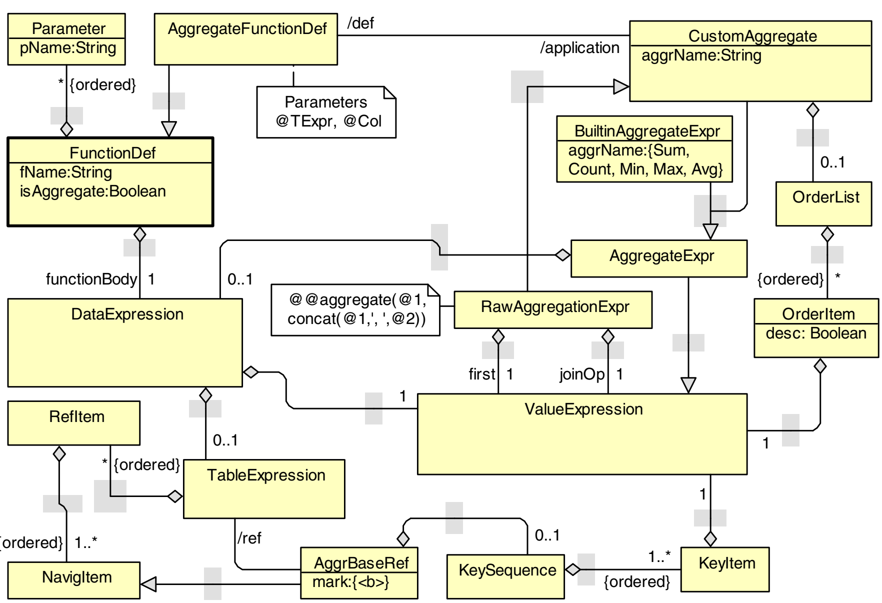


Fig. 8. RDB2OWL Aggregate Function metamodel

thought of as a starting point from which table reference or navigation list of E are started to get to the table in which the value expression *V* is evaluated as an argument for the aggregate function *f*. For example, to calculate total salary for a person where Person-to-Work tables are in 1:n relation one would write datatype property map expression in one of the forms:

```
Person.Sum(<b>=>Work.Salary)
Person.Sum(=>Work.Salary)
Person.Sum(
(<b> {key=(PersonID)}, Work w;
<b>.PersonID=w.PersonID).Salary )
```

In this example *Person* is the base table referenced by <b> (omitted in the short form). The longest form shows the use of explicit key sequence (*KeySequence, KeyItem* instance) that specifies grouping by option. When key sequence is omitted, the primary key column sequence for base table expression is assumed. The above example expression can translate into an execution environment as:

```
SELECT sum(Salary)
FROM Person p, Work w
WHERE p.PersonID=w.PersonID
GROUP BY p.PersonID
```

In the mini-University example (recall Fig 1 and Fig 2), to calculate the course count that each teacher teaches, one can use *[[Teacher]]* notation to refer to the sole class map for the *Teacher* class, thus writing: *[[Teacher]].Count(<b> {key=(AutoID)} => XCourse.AutoID)*

RDB2OWL has built-in function *@@aggregate* (*RawAggregationExpr* instance) offering custom aggregate expression definitions. *@@aggregate* takes 4 arguments:

– a table expression, including a reference to the <b>-tagged context expression and a defined key list (within the <b>-tagged expression), and optionally an order by clause;
– a value expression to be aggregated over
– a single argument function for first value processing in the aggregate formation (the sole variable for this function is denoted by @1)
– a two argument function for adding the next value to the aggregate (the value accumulated so far is denoted by @1 and the next value is denoted by @2).

For example, to get the course list (comma-separated code list) each student is registered to, one would write: *[[Student]].@@aggregate((<b>=>XRegistration-> XCourse {Code asc}), Code, @1, #concat(@1, ‘, ‘, @2)).*

User defined aggregate functions (*AggregateFunctionDef* instance) can be defined with *@@aggregate* function. If variables named @TExpr (denotes a table expression) and @Col (denotes columns for value expression) are present in the context of the call to *@@aggregate*, the first two arguments in the call may be omitted, they are filled by the values of these variables. This allows shorter forms of user-defined aggregate function definition (‘@@’ is the name prefix for user-defined aggregate functions):

@@List(@TExpr,@Col) = @@aggregate(@1, #concat(@1, ', ', @2))

Shorter forms of aggregate function definition omit variables *@TExpr* and *@Col*:

@@List()= @@aggregate(@1, #concat(@1, ', ', @2)).

To get course list one can apply this user-defined aggregate function @@List:

[[Student]].@@List((=>XRegistration->XCourse {Code asc}).Code)

In this example the table expression *=>XRegistration->XCourse {Code asc}* is assigned to variable @*TExpr* and the value expression *Code* – to variable @*Col*.

3.5 Extended Mapping Example

We present an example illustrating the advanced RDB2OWL construct application.

Figures 9 and 10 show extended mini-University DB schema and target ontology example with mapping annotations. Ontology level annotations describe two database schemas- one for source database (referenced by 'M') and auxiliary database (referenced by 'A') for which SQL script *RDB2OWL_init* is specified to be executed before start of mapping processing for triple generation. The list of user-defined function definitions is located also in ontology level annotations. Note that definition for *split* function references auxiliary database A where auxiliary table Numbers resides. This function *split* splits a coma separated value into its parts, e.g., '11,12,13'→{'11','12','13'}, its definition uses another RDB2OWL function *encoma* that puts comas around string value ('11'→',11,').

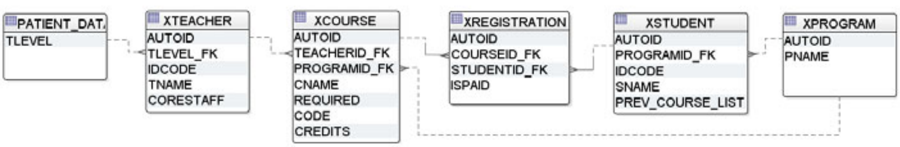


Fig. 9. An extended mini-university database schema

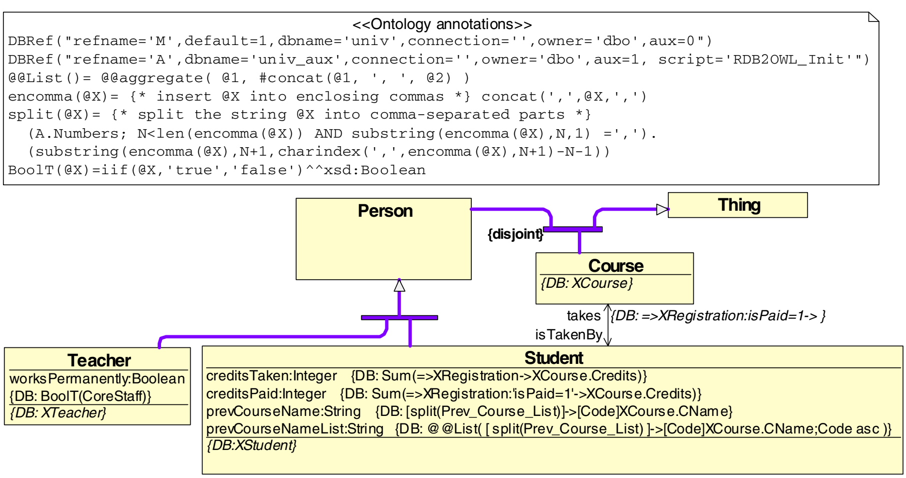


Fig. 10. Extended ontology example

Aggregate built-in function *Sum* is applied to define datatype property map for property *creditsTaken*. Because expression *Sum(=>XRegistration->XCourse.Credits)* omits an explicit base table expression it is assumed to be the one defined for the sole class map of *Student* class which is *XStudent*. Expanded form would be *XStudent.Sum(<b>{key=(AutoID)}=>XRegistration->XCourse.Credits).* Aggregate expression for *creditsPaid* property is defined similarly; it uses also row filtering condition.

In this extended mini-University example table *XStudent* has column *prev_course_list* to hold comma-separated list of codes of previous course list, e.g., 'semweb,prog01,prog02,softeng' (not a good database design but ontologies should be map-able to real databases). In ontology *prevCourseName* property is with cardinality larger than 0. The mapping expression for this property *[split(Prev_Course_List)]->[Code]XCourse.CName* specifies 2-step transformation:

1) *split(Prev_Course_List)* means splitting of comma-separated value in *prev_course_list* into separate parts: 'semweb,prog01,prog02,softeng'→{'semweb', 'prog01', 'prog02', 'softeng'},

2) separate value list from step (1) is put into navigation link structure as column value to join with *XCourse* table to get name list from code list, e.g., *[{'semweb', 'prog01', 'prog02', 'softeng'}]->[Code]XCourse.CName→{'Semantic Web', 'Programming 1', 'Programming2', 'Software Engineering'}.*

Datatype property map expression for property *prevCourseNameList* is bit more complicated: the same expression as for property *prevCourseName* is put as argument in application of user defined @@*List* function to obtain the list of previous course names into comma-separated string.

4 Conclusions

We have presented advanced features of RDB2OWL approach for RDB-to-RDF/OWL mapping specification that are written as OWL ontology annotations. Convenience of mapping definitions has been illustrated by example. Power of RDB2OWL combined with visual ontology modeling tools such as OWGrEd [23] can be viable mechanism for RDB semantic re-engineering. An annotated OWL ontology with RDB mappings can be thought also as a documentation describing the technical data schema from the perspective of the conceptual model (ontology).

The RDF triple generation on the basis of an intermediate RDB-to-RDF mapping encoding within a relational database schema (the RDB2OWL mapping DB schema) has been successfully implemented in real life case of semantic re-engineering of Latvian Medical registry databases (42,8 million triples have been generated in 20 minutes from mappings stored in special RDB schema). On the other hand, the Latvian Medical registry ontology has been successfully annotated with RDB2OWL annotations (the aggregate functions were not needed in the defined mapping case). Implementation of full set of RDB2OWL constructs is in progress including syntax level parsing, syntax model transformation to semantic model and to the intermediate execution model (RDB2OWL mapping DB schema).

Another interesting venue of research would be translation of RDB2OWL mappings into more commonly used RDB-to-OWL mapping formalisms, such as D2RQ or Virtuoso RDF Views, to enable on-the-fly access to the relational databases from a RDF/SPARQL endpoint, on the basis of specified RDB2OWL mappings.

References

1. Resource Description Framework (RDF), `http://www.w3.org/RDF/`
2. SPARQL 1.1 Query Language, `http://www.w3.org/TR/2010/WD-sparql11-query-20100601/`
3. OWL 2 Web Ontology Language, Structural Specification and Functional-Style Syntax, `http://www.w3.org/TR/2009/REC-owl2-syntax-20091027/`
4. RDF Vocabulary Description Language 1.0: RDF Schema, `http://www.w3.org/TR/rdf-schema/`
5. Barrasa, J., Corcho, O., Shen, G., Gomez-Perez, A.: R2O: An extensible and semantically based database-to-ontology mapping language. In: SWDB 2004, 2nd Workshop on Semantic Web and Databases (2004)
6. D2RQ Platform, `http://www4.wiwiss.fu-berlin.de/bizer/D2RQ/spec/`
7. Blakeley, C.: RDF Views of SQL Data (Declarative SQL Schema to RDF Mapping), OpenLink Software (2007)
8. Hu, W., Qu, Y.: Discovering Simple Mappings Between Relational Database Schemas and Ontologies. In: Aberer, K., Choi, K.-S., Noy, N., Allemang, D., Lee, K.-I., Nixon, L.J.B., Golbeck, J., Mika, P., Maynard, D., Mizoguchi, R., Schreiber, G., Cudré-Mauroux, P. (eds.) ASWC 2007 and ISWC 2007. LNCS, vol. 4825, pp. 225–238. Springer, Heidelberg (2007)
9. Sequeda, J.F., Cunningham, C., Depena, R., Miranker, D.P. Ultrawrap: Using SQL Views for RDB2RDF. In: Poster Proceedings of the 8th International Semantic Web Conference (ISWC 2009), Chantilly, VA, USA. (2009)
10. Auer, S., Dietzold, S., Lehmann, J., Hellmann, S., Aumueller, D.: Triplify: Light-weight linked data publication from relational databases. In: Proceedings of the 18th International Conference on World Wide Web (2009)
11. R2RML: RDB to RDF Mapping Language, `http://www.w3.org/TR/r2rml/`
12. A Direct Mapping of Relational Data to RDF, `http://www.w3.org/TR/2011/WD-rdb-direct-mapping-20110324/`
13. de Laborda, C.P., Conrad, S.: Database to Semantic Web Mapping Using RDF Query Languages. In: Embley, D.W., Olivé, A., Ram, S. (eds.) ER 2006. LNCS, vol. 4215, pp. 241–254. Springer, Heidelberg (2006)
14. Bizer, C., Schultz, A.: The R2R Framework: Publishing and Discovering Mappings on the Web. In: 1st International Workshop on Consuming Linked Data (COLD 2010), Shanghai (November 2010)
15. Rikacovs, S., Barzdins, J.: Export of Relational Databases to RDF Databases: A Case Study. In: Forbrig, P., Günther, H. (eds.) BIR 2010. LNBIP, vol. 64, pp. 203–211. Springer, Heidelberg (2010)
16. Būmans, G., Čerāns, K.: RDB2OWL: Mapping Relational Databases into OWL Ontologies - a Practical Approach. In: Databases and Information Systems: Proc. of the Ninth International Baltic Conference, Baltic DB&IS 2010, Latvia, Riga, July 5-7, pp. 393–408. Riga, University of Latvia, Riga (2010)

17. Čerāns, K., Būmans, G.: RDB2OWL: a RDB-to-RDF/OWL Mapping Specification Language, vol. 224, pp. 139–152. IOS Press, Amsterdam (2011)
18. Bumans, G.: Mapping between Relational Databases and OWL Ontologies: an Example. To Appear in Scientific Papers of University of Latvia, Computer Science and Information Technologies (2010)
19. Būmans, G., Čerāns, K.: RDB2OWL: a Practical Approach for Transforming RDB Data into RDF/OWL. In: Proceedings of the 6th International Conference on Semantic Systems, Graz, Austria, ACM International Conference Proceeding Series, Article No.25 (September 2010); ISBN 9781450300148
20. Barzdins, G., Liepins, E., Veilande, M., Zviedris, M.: Semantic Latvia Approach in the Medical Domain. In: Haav, H.M., Kalja, A. (eds.) Proc. 8th International Baltic Conference on Databases and Information Systems, pp. 89–102. Tallinn University of Technology Press (2008)
21. Barzdins, G., Rikacovs, S., Veilande, M., Zviedris, M.: Ontological Re-engineering of Medical Data-bases. Proc. of the Latvian Academy of Sciences, Section B 63(4/5) (663/664), 20–30 (2009)
22. Barzdins, G., Barzdins, J., Cerans, K.: From Databases to Ontologies, Semantic Web Engineering in the Knowledge Society. In: Cardoso, J., Lytras, M. (eds.) IGI Global, pp. 242–266 (2008) ISBN: 978-1-60566-112-4
23. OWLGrEd, `http://owlgred.lumii.lv/`
24. Moden, J.: The "Numbers" or "Tally" Table: What it is and how it replaces a loop, `http://www.sqlservercentral.com/articles/T-SQL/62867/`

Export of Relational Databases to RDF Databases by Model Transformations

Sergejs Rikacovs

Institute of Mathematics and Computer Science, University of Latvia
Raina blvd. 29, LV-1459, Riga, Latvia
Sergejs.Rikacovs@lumii.lv

Abstract. The Semantic Web is a Web of Data. To fulfill this web with data we need methods how to transfer business data from existing relational databases. In most cases, textual mapping languages are used for the specification of correspondences between relational DB schema and OWL ontology. These languages generally are rather awkward and not well-suited for the specification of mappings in cases when there is a substantial semantic gap between the source ER schema and the target OWL ontology. At the same time specification of mappings is a classical use case for graphical model transformation languages. In our previous work [10] we have proposed a new, model transformation-based method for the specification of correspondences between the elements of DB schema and OWL ontology. During practical approbation of this method it became clear that the proposed method can be improved with respect to RAM usage and execution performance. In this paper we describe improvements of the previously proposed method that allows us to overcome these problems.

Keywords: OWL, RDF, relational database, migration.

1 Introduction

The Semantic Web is visioned as a Web of Data. However, nowadays the vast amount of business information is still locked in relational data bases. To make the vision of the Semantic Web a reality we need ways how to populate it with data from the existing relational databases. At the given point this problem is topical both from research and application point of view.

There are several requirements that are very important in this context:

1. We need a language that allows us to specify both trivial and sophisticated mappings between the source ER schema and the target OWL ontology
2. This language must have an efficient implementation
3. Mappings specified in this language should be both easily writable and easily comprehensible

It is quite obvious that these requirements are conflicting, and certain compromises should be found.

At the given point, there already exist several methods for migration of relational data to RDF. A survey of the existing approaches is given in the W3C work group

J. Grabis and M. Kirikova (Eds.): BIR 2011, LNBIP 90, pp. 158–166, 2011.

report [4].These methods can be divided into two groups: the ones providing a direct creation of RDF dumps and the others allowing to define views on relational data [5, 6, 7,11].

One of the most mature implementations of mapping languages are Virtuoso RDF View [6] and D2RQ [5]. Virtuoso RDF View provides its own textual Quad map language that allows to define RDF views on relational data. A potential drawback of this language is the absence of a strict control of the target ontology. It is not prohibited to map some relational table to a nonexistent element of ontology. This can result in some very obscure bugs. Despite the fact that mappings expressed using the Quad map language can be quite difficult to comprehend the execution performance is quite impressive.

D2RQ provides a declarative language for defining RDF views on relational data. The main strength of mapping languages is in the specification of mappings for typical situations. However, there are situations when a limited expressive power of mapping languages makes the specification of sophisticated mappings rather difficult or even impossible (without using some SQL programming).

As to model based approaches to migration of relational data to RDF, a quite interesting approach is discussed in [3] where the authors provide a mapping language that is compiled to a model transformation language. In a certain way this approach is relatively similar to ours. The main differences lie in the fact that we are using a graphical model transformation language (that makes it easier for domain experts to specify and understand the transformation rules).

In [10] we have proposed a new, model transformation-based method for the specification of correspondences between the elements of DB schema and OWL ontology. The main advantage of the proposed method is easily readable, easily understandable and easily writable transformations specifying mappings between RDB and RDF. According to this method the process of data migration consists of the following steps:

1. We start by importing data and ER schema of the source relational database into a meta-model based data store [9]
2. Then, we import target ontology (OWL ontology) into the meta-model based data store
3. Then, a domain expert specifies correspondences between the elements (a group of elements) of the source ER model and corresponding elements of the target ontology. These correspondences are specified in a high-level graphical model transformation language MOLA [2].
4. After execution of transformations created in the previous step, NTriple files are generated according to the rules specified by a domain expert. These files contain the representation of relational data in the form of RDF triples.
5. Finally, newly created NTriple files are imported into the RDF database.

It should be stressed that all of the steps listed above, excluding only one step – the specification of correspondences between the source ER model and the target ontology, are universal steps, as they do not depend on a concrete source ER model or target OWL ontology or correspondences between them. It means that the only step that needs to be taken care of by the user of this method is the specification of correspondences between the source ER model elements and the elements of the target OWL ontology.

This method has also been approbated on a real world use case by successfully migrating relational databases of 6 Latvian medical registries [1] into a single shared RDF database [10]. Migration was successfully finished, but during this practical approbation it became clear that the proposed method can be improved with respect to RAM usage and execution performance.

For instance, to migrate aforementioned relational databases of 6 Latvian medical registries containing 3 Gb of data we needed about 8 Gb of RAM. The reason for such an extensive RAM consumption is related to the fact that the model transformations performing migration of data are implemented on top of an in-memory data store. One more problem stemming from the usage of an in-memory data store is the necessity to import source relational data to an in-memory data store before mappings that specify correspondences between the source ER schema and the target OWL ontology can be executed. In case of the aforementioned migration of medical data bases this import took about 1 hour and 30 minutes. The execution performance on data bases of substantial size could be better as well.

In this paper we try to overcome problems related to the use of an in-memory data store rather radically – we use relational DBMS instead of an in-memory data store for the implementation of our method. In this case DBMS takes care of memory management, and the problem of excessive RAM usage disappears. The problem of data import to an in-memory data store also becomes obsolete - data are transformed in place (in a relational database). Next section gives a more detailed description of the above mentioned ideas.

2 Model Transformation-Based Migration Method

The conceptual schema of this method is given in Fig. 1. According to this schema the migration process consists of the following steps:

1. Import target OWL ontology into the database. During this import we create relational tables, representing target OWL ontology and its instances.
2. Specify mappings between the source class diagram (automatically obtained from source relational schema) and the target OWL ontology. These correspondences are specified in a high level graphical model transformation language MOLA.
3. Compile MOLA program to SQL.
4. Execute the obtained SQL code – this code populates relational tables representing OWL ontology, with data to be exported as prescribed by MOLA mapping specified in the previous step.
5. Export data, contained in relational tables, representing OWL ontology to RDF files.

It should be noted that all of these steps, except the specification of mappings that requires domain knowledge and is dependent on a specific pair of source ER schema and target OWL ontology are fully universal (in a sense that they are not dependent on the target ontology or the source ER schema) and do not require any human involvement. One of the corner stones of this method is the transformation of one MOF [12] metamodel instance to another metamodel instance. This problem is also quite well-known in a classical MDA approach, where the Platform Independet Model

(PIM) is transformed to a Platform Specific Model (PSM). It is a classical use case for model transformation languages with the only difference that the metamodel and its respective instances are stored in a database. But, usually model transformation languages are implemented on top of some metamodel based data store, and not on top of a relational data base.

Therefore it is vitally important to implement an efficient compiler of MOLA to SQL. Our first impression was that this kind of efficient compiler will be rather difficult to implement – because SQL supports efficient manipulation on the level of tables (or "big data chunks") but MOLA operates on the level of individual objects. So here we had 2 options:

- Compile MOLA to row-by-row SQL operations (instead of set operations).
- Restrict MOLA expressivity in such a way that only class level processing would be allowed (processing of individual objects would be prohibited).

We choose the first option, and therefore the main goal of this paper is to verify how the proposed model-transformation based migration method will perform from the efficiency point of view.

Traditionally, model transformation languages are implemented by compiling high level transformation language to lower level base transformation language and finally compiling base transformation language to executable code [8].

In case of MOLA the base transformation language is L0 [14], and there already is an efficient MOLA to L0 compiler [8] implemented through a bootstrapping method [15]. That is why in this paper we assume that transformation program is specified in L0, and we concentrate on the efficient compilation of L0 program to SQL.

2.1 Accessing Source ER Schema

To implement the above mentioned ideas we need to transform a database ER schema to a UML class diagram. It is quite simple:

1. Every table T from the source relational schema is represented with a class C. Every record R of table T is represented with an object O of class C.
2. Every column *Col* of a table *T*, excluding FK columns and PK columns, is represented by an appropriate class attribute *A*. Values of column *Col*, are represented by values of attribute *A*.
3. Every foreign key relation is represented with an association between appropriate classes.

2.2 Importing Target OWL Ontology

To provide a way to process target OWL ontology with corresponding instances, an ER schema, representing target OWL ontology, is created in a data base that contains source relational data.

In general, representation of OWL ontology with relational tables can be a rather non-trivial task [13]. However, we confine ourselves to only those ontologies that conform to UML/OWL subset. The basic idea of this subset is to use only those OWL DL constructs that can be adequately represented with UML class diagrams.

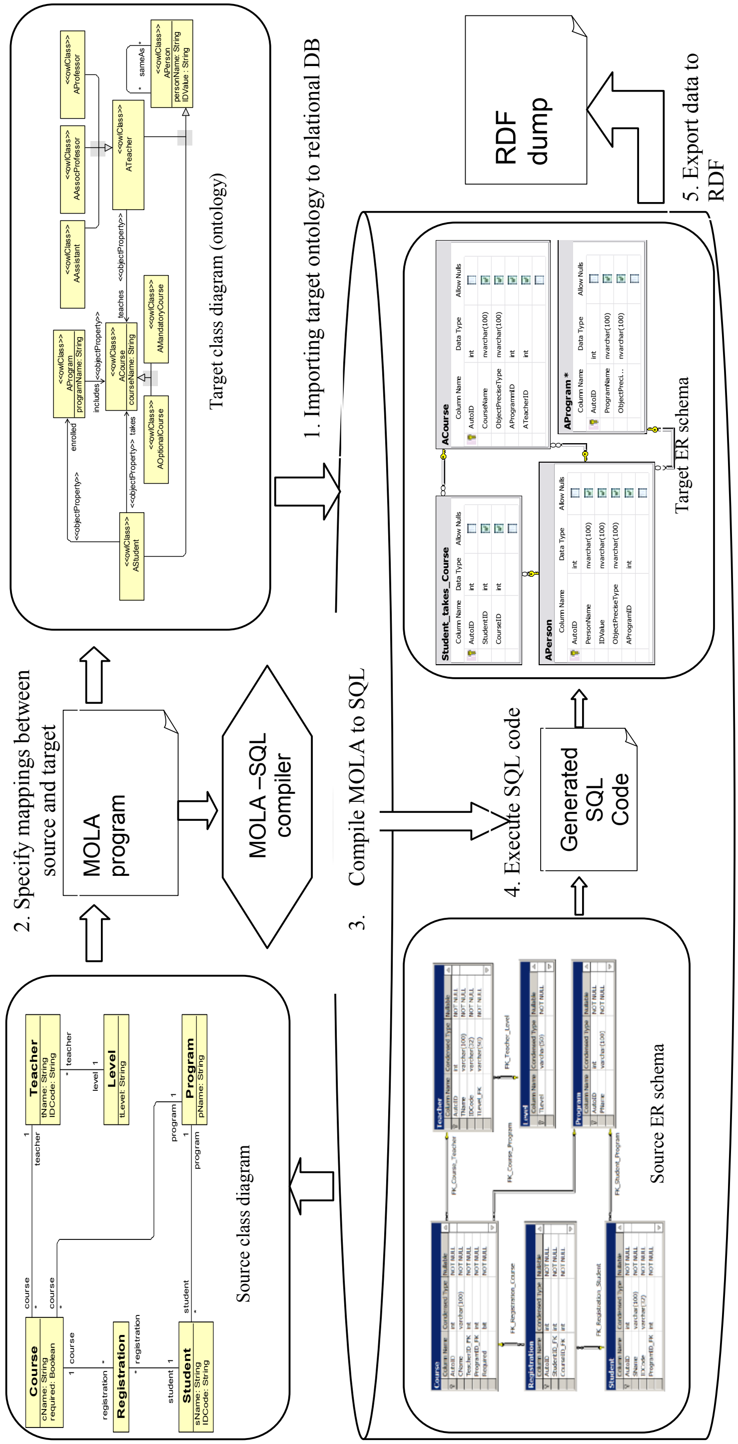


Fig. 1. The conceptual schema of the proposed method

If ontology conforms to a UML/OWL subset, we can derive quite a simple set of rules for importing the ontology into a relational database:

- For every <<owlClass>>, a corresponding relational table is created.
- For every owl <<objectProperty>>, a foreign key relation between appropriate tables is created. If the object property represents an m:n relation, a junction table is created.
- For every owl <<datatypeProperty>>, a table column of appropriate type is created (at the given point we support only four elementary types – real, boolean, string, integer).
- Inheritance hierarchies are flattened. It means that we create only one relational table corresponding to the root of this hierarchy and all classes derived from this root will be represented by the same table (special descriptors are used to denote the precise object type).

Now we can freely manipulate elements of this ontology through a model transformation program.

2.3 Exporting RDF Triples

The last step of the migration process is the generation of NTriples from instances contained in relational tables representing OWL ontology:

- For every row of every table, we generate triples in the form <ObjURI> <rdf:Type> <ClassURI>.
- For every link between objects (be it a simple foreign key relation or a junction table), we generate triples in the form <ObjURI1> <ObjPropURI> <ObjURI2>.
- For every column value present in the relational tables representing OWL ontology, we generate triples in the form <ObjURI> <DataTypeURI> <AttributeValue>.

2.4 Main Step: Compilation of L0 to SQL

L0 language is a textual low-level model transformation language. Its constructs can be divided into following groups:

- Model(both objects and links) iteration commands
- Model update commands
  - creation/deletion of objects and links
  - getting/setting the value of an attribute of an object
- Control flow commands
  - low level control flow instructions and labels

All commands are executed on the level of an individual object. There are no commands that allow set processing [14].

We start the explanation of compilation principles with model iteration commands. The most straightforward approach for iterating through SQL table rows are SQL cursors. However, there are also some penalties for using them. The main one being performance problems. That is why we choose to compile L0 model iteration commands to row-by-row SQL operations without using cursors. The main idea of this compilation is to represent L0 pointer by several SQL variables representing the state of it. If a pointer is used to iterate through class objects, the following information about the pointer state is required:

- initialization kind – shows what L0 command was used to initialize the pointer (there are 3 options: *first*, *first from by*, *first from where*)
- current row id – represents id of the current row to be processed
- next row id – represents the next row to be processed

In case of iteration through association instances-links (*first from by*) and iteration through objects satisfying given conditions we need more information about the pointer state, but here we will omit these details.

Table 1. Principles of L0 command compilation

First <pointerName> : <Class> **else** <Label>;	**SET** @<pointerName >_InitKind = 1; **SET** @next_<pointerName >_RowID = NULL; **SELECT** @next_<pointerName >_RowID = **MIN**(GetTablePK(<Class>)) FROM <Class>; **IF ISNULL**(@next_<pointerName >_RowID,0) = 0 **BEGIN GOTO** <Label>; **END ELSE BEGIN** **SET** @curr_<pointerName>_RowID = @next_<pointerName>_RowID; **END;**
Next <pointerName> **else** <Label>;	**IF** @<pointerName>_InitKind = 1 **BEGIN** **SET** @next_<pointerName>_RowID = **NULL** **SELECT** @next_<pointerName>_RowID = **MIN**(<referencedClassName>.GetTablePK(<referencedClassName>) **FROM** <referencedClassName> **WHERE** <referencedClassName>.GetTablePK(<referencedClassName>) > @curr_<pointerName>_RowID; **IF ISNULL**(@next_<pointerName>_RowID,0) = 0 **BEGIN GOTO** <Label>; **END ELSE BEGIN**; **SET** @curr_<pointerName>_RowID = @next_<pointerName>_RowID; **END**; **END**;
addObj <pointerName>:<ClassName>;	**INSERT INTO** [<className>] default values; **SET** @curr_<pointerName>_RowID = **SCOPE_IDENTITY**(); **SET** @next_<pointerName>_RowID = @curr_<pointerName>_RowID;
addLink <pointer1Name>.<roleName>.<pointer2Name>;	**UPDATE** GetReferencedClassName(<pointer1Name>) **SET** <roleName> = @curr_<pointer2Name>_RowID **WHERE** GetReferencedClassName(<pointer1Name>). GetTablePK(GetReferencedClassName(<pointer1Name>)) = @curr_<pointer1Name>_RowID;
label <labelName>;	<labelName>:
goto <labelName>;	**GOTO** <labelName>;

Model update operations are translated to SQL INSERT and UPDATE sentences modifying rows representing corresponding elements of the target ontology. As to control flow commands, its compilation is rather straightforward: L0 goto command and labels are compiled to its direct counterparts in SQL. Other control flow instructions are compiled to a related constructs of SQL. In Table 1 an illustration of these principles is given.

3 Results of Practical Application

We have implemented a prototype of L0-SQL compiler with respect to the principles specified above and compared the performance of the migration process when it is implemented on top of an in-memory data store and when it is implemented on top of relational data base (i.e. transformations are executed in place). At the given moment we have compared the migration of only one relational database - System Core - instead of 6 medical databases as it was in the original setting. There were several reasons for this. Firstly, it is easier to test migration of one DB. Secondly, in our prototype implementation of L0-SQL compiler some advanced transformation language features (derived class processing, m:n association processing, connecting multiple types to an object) have not been implemented yet, but these features are used for migration of the remaining 5 DB. It should be stressed that it is definitely clear how to implement them. According to this comparison, the migration method implemented on top of relational DB demonstrates approximately 2 times better performance. Problems concerning excessive RAM usage (DBMS takes care of memory management) and the necessity to import relational data have also been overcome.

4 Conclusions and Future Work

Our first experiments have already showed that model transformation based migration method proposed in this paper is practically applicable, and it is approximately 2 times more efficient from performance point of view than the method described in our previous paper. Moreover, the method proposed in this paper does not demand special requirements on the amount of the required RAM and does not force the user to import relational data to be transformed into an external in-memory data store.

Our future plans include further development of the proposed method and its supplementation with yet unimplemented features.

Acknowledgments. This work has been supported by the European Social Fund within the project “Support for Doctoral Studies at University of Latvia”.

References

1. Barzdins, G., Rikacovs, S., Veilande, M., Zviedris, M.: Ontological Re-engineering of Medical Databases. Proceedings of the Latvian Academy of Sciences. Section B, 63(4/5) (663/664), 156–158 (2009)
2. Kalnins, A., Barzdins, J., Celms, E.: Model Transformation Language MOLA. In: Aßmann, U., Aksit, M., Rensink, A. (eds.) MDAFA 2003. LNCS, vol. 3599, pp. 62–76. Springer, Heidelberg (2005)

3. Hillairet, G., Bertrand, F., Lafaye, J.-Y.: MDE for Publishing Data on the Semantic Web. In: Proceedings of the First International Workshop on Transforming and Weaving Ontologies in Model Driven Engineering (TWOMDE 2008), pp. 32–46 (2008)
4. Sahoo, S.S., Halb, W., Hellmann, S., Idehen, K., Thibodeau Jr, T., Auer, S., Ezzat, A.: A survey of current approaches for mapping of relational databases to RDF. Technical report, W3C RDB2RDF Incubator Group (January 2008)
5. D2RQ Platform, `http://www4.wiwiss.fu-berlin.de/bizer/D2RQ/spec/`
6. Blakeley, C.: RDF Views of SQL Data (Declarative SQL Schema to RDF Mapping), OpenLink Software (2007)
7. Hu, W., Qu, Y.: Discovering Simple Mappings Between Relational Database Schemas and Ontologies. In: Aberer, K., Choi, K.-S., Noy, N., Allemang, D., Lee, K.-I., Nixon, L.J.B., Golbeck, J., Mika, P., Maynard, D., Mizoguchi, R., Schreiber, G., Cudré-Mauroux, P. (eds.) ASWC 2007 and ISWC 2007. LNCS, vol. 4825, pp. 225–238. Springer, Heidelberg (2007)
8. Sostaks., A., Kalnins, A.: The implementation of MOLA to L3 compiler, Scientific Papers University of Latvia. Computer Science and Information Technologies (2008)
9. Opmanis, M., Cerans K.: JR: A Multilevel Data Repository. In: Proceedings of the 9th International Baltic Conference on Databases and Information Systems (Baltic DB&IS 2010), Riga, Latvia, July 5-7, pp. 375–390 (2010)
10. Rikacovs, S., Barzdins, J.: Export of Relational Databases to RDF Databases: A Case Study. In: Forbrig, P., Günther, H. (eds.) BIR 2010. LNBIP, vol. 64, pp. 203–211. Springer, Heidelberg (2010)
11. Čerāns, K., Būmans, G.: RDB2OWL: a RDB-to-RDF/OWL Mapping Specification Language. In: Barzdins, J., Kirikova, M. (eds.) Proceeding of the 2011, Conference on Databases and Information Systems VI: Selected Papers from the Ninth International Baltic Conference, DB&IS 2010, pp. 139–152. IOS Press, Amsterdam (2011)
12. Request for Proposal:MOF 2.0 Query / Views / Transformations RFP, `http://www.omg.org/docs/ad/02-04-10.pdf`
13. Vysniauskas, E., Nemuraite, L., Sukys, A.: A Hybrid Approach for Relating OWL 2 Ontologies and Relational Databases. In: Forbrig, P., Günther, H. (eds.) BIR 2010. LNBIP, vol. 64, pp. 86–101. Springer, Heidelberg (2010)
14. Rikacovs, S.: The base transformation language L0+ and its implementation, Scientific Papers, University of Latvia. Computer Science and Information Technologies, Latvia, Riga, pp. 75–102 (2008)
15. Barzdins, J., Kalnins, A., Rencis, E., Rikacovs, S.: Model Transformation Languages and their Implementation by Bootstrapping Method. LNCS, pp. 130–145. Springer, Heidelberg (2008)

An Analysis of Enterprise Architecture Maturity Frameworks

Martin Meyer[1], Markus Helfert[1], and Conor O'Brien[2]

[1] Dublin City University, Dublin Ireland
{mmeyer,markus.helfert}@computing.dcu.ie
[2] Innovation Value Institute,Maynooth Ireland
conor.obrien@nuim.ie

Abstract. Enterprise Architecture (EA) has gained a lot of attention in literature and industry in recent years since it serves as valuable instrument to guide the enterprise through the transformation from a current to a future state by means of providing standardization, rules and principles. This ensures currently the best known way to achieve flexibility, reusability and a common understanding of the enterprise and its functions within the IT domain. The evolvement of EA is best captured by employing a maturity model which indicates the status quo of the EA and provides a means of further improving this evolvement. Several frameworks and approaches have been proposed throughout the years and we conducted an analysis of selected of EA Maturity frameworks in terms of five – in our opinion crucial - key characteristics. Moreover, as EA covers the enterprise IT holistically on a high level of abstraction, we found an overarching framework available for assessing all of the critical IT functions.

Keywords: Enterprise Architecture, Maturity Model, Maturity Framework Analysis.

1 Introduction

Capturing the complexities of today's enterprises poses an immense challenge since it involves in-depth knowledge of the organizations, processes, and stakeholders as well as their relations to each other. This is especially true for large enterprises. To support the documentation and management of enterprise IT, many frameworks have been proposed from industry and academia. This is where we enter the realm of Enterprise Architecture (EA) which is the *"the fundamental organization of a system embodied in its components, their relationships to each other, and to the environment, and the principles guiding its design and evolution."* [1].

Companies adapting EA in their enterprise functions need to be aware of the scope and impact by doing so. Furthermore, EA must continuously evolve and therefore we need to somehow measure the progress. For this purpose, the concept of maturity was employed for EA which assigns different levels of achievement by means of a maturity assessment to artifacts, processes, or characteristics respectively. These levels indicate how advanced such entities are in their current stage of evolvement. In the end, a higher maturity is sought after in order to increase the value creation from

J. Grabis and M. Kirikova (Eds.): BIR 2011, LNBIP 90, pp. 167–177, 2011.

IT assets. The aim of this paper is to provide a scheme to analyze such EA Maturity models, frameworks or approaches by means of adequate key characteristics.

The rest of this work is organized as follows: Section 2 covers EA Maturity after which we have our selection of EA Maturity frameworks and inspect their scope in section 3. Then, in section 4 we provide the framework analysis and finally conclude and give directions for future research in section 5 and 6 respectively.

2 Enterprise Architecture Maturity

Maturity in the IT domain is often seen as part of Quality Management. More mature solutions possess better quality in terms of operational efficiency. Typically, maturity models are designed for a specific domain in order to measure the current state the achieved level of competence by means of a maturity assessment [2]. Consequently, when speaking of maturity we refer to a measure. An EA Maturity level is a value obtained through the aggregation of assessing different enterprise components. EA Maturity models support the improvement of the EA domain. Assessing EA maturity still poses a great challenge for industry and academia. It is critical to choose the adequate characteristics of a good EA and how to match those [3]. Specific EA maturity approaches are still a scarce resource in literature and therefore there is not a common definition available [4]. EA Maturity, as an approach, delivers a measure to indicate the enterprises' current stage of development in terms of IT capabilities relevant for the scope of EA.

2.1 EA Maturity Objectives

The purpose of maturing the EA is the same as maturing any other domain for which a maturity model was developed. It is to increase its performance and effectiveness upon achieving a higher maturity. The objectives which are sought-after by enterprises when using maturity models or frameworks respectively can be summarized as follows:

- Increase in performance, effectiveness, efficiency, and value generation in terms of planning, development, and operation according to the strategy
- Decrease the expenditure of costs and time in terms of development and operation
- Obtain better understanding and knowledge of the enterprise and its structures as well as their evolvement, e.g. the organizational structure and the corresponding communications

From these objectives, we can derive three perspectives or dimensions respectively containing several focus areas which are in general interest for companies when using a maturity approach:

- *Strategy:* Sets the direction of the business and IT in a top-down approach, e.g. Strategic Planning, Finance, Governance
- *Architecture*[1]*:* Comprises the Business-, Application-, and Technology Architecture

[1] The Information Architecture is part of the Application Architecture.

- *Operations:* The daily work with the IT systems, e.g. Project Management, IT Service Management, Quality Management

Within these perspectives, maturity frameworks leave their footprint by assessing the status quo and providing the necessary steps in order to improve the maturity. We will use these perspectives in the course of our scope analysis in section 3.1.

2.2 EA Maturity Method

The EA Maturity method determines the structure, roles, activities and results of the maturity assessment, i.e. or in other words how to use the maturity model in order to exploit its sought-after benefits. Several types of assessment methods can be found in literature for various maturity models. In most cases, the maturity level is determined by a Likert-like questionnaire, i.e. the actual level assignment is done on a textual basis. Other ways include expert interviews or quantitative analysis, and combinations thereof.

2.3 EA Maturity Model Types

A possible way to differ between architecture maturity models is given in [5] where three types are identified upon number of levels and focus areas. In our opinion, a core feature of every maturity model is the assignment of maturity levels, and in some cases, even capability levels [6]. The important question is now to which enterprise objects, such as processes, organizations, or structures, or, more abstractly, to which characteristic, category or capability the maturity level is assigned. From the major maturity models we have examined, we can derive two different types of EA maturity models. *Process-based* EA Maturity frameworks assign maturity levels to processes within the enterprise. Hence, they measure a set of activities performed by various stakeholders. When frameworks are centered on processes, they usually provide an in-depth description of these including inputs, outputs and goals. *Characteristics-based* EA Maturity assesses different characteristics, criteria, categories, or attributes respectively. Although some characteristics-based frameworks consider processes as distinct characteristic, we refer to it as a set of attributes which describe the domain and focus area in an appropriate way. In contrast to the process-based types, the assessment is unlike more difficult, since processes are often well supported by software applications and therefore easier to govern and monitor, especially in terms of quantitative evaluation.

3 EA Maturity Models, Frameworks and Approaches

We will now take a look at a selection of frameworks developed for supporting the evolvement of EA and which are part of our framework analysis. The selection is summarized in Table 1.

Further maturity models can be found in industry and literature, e.g. an algorithm-based maturity model which generates the model out of questionnaire data which

Table 1. Selected EA Maturity frameworks

Framework	Publisher	Type
IT-CMF [7]	IVI	Process-/Characteristics-based
EAMM [8]	NASCIO	Characteristics-based
SAMM [9][10]	Luftmann	Characteristics-based
CMMI [11]	SEI Carnegie Mellon University	Process-based
ACMM [12]	U.S. DoC	Characteristics-based
EAMMF/EAAF [13][14]	U.S. GAO, U.S. OMB	Characteristics-based
COBIT/ValIT [15][16]	ITGI	Process-based

follows an inductive design approach [17]. The assignment of maturity levels is still done for different characteristics and processes. In [5], a matrix-based maturity model defining 18 architecture practice focus areas is presented, which supports up to 13 maturity levels to allow a finer level of detail during assessments. Bringing alignment and maturity for architecture together in a multi-dimensional model is done in [18] by identifying six key variables that explain them both. The *IT Infrastructure Library (ITIL)* [19] is the de-facto standard for IT Service Management. It offers the *Process Maturity Framework (PMF)* which is aligned with the *CMMI*.

Table 2. Scores for the EA Maturity Framework scope

	Frameworks						
Dimension (Focus Areas)	*EAMM*	*SAMM*	*CMMI*	*ACMM*	*EAMMF*	*COBIT/ ValIT*	*IT-CMF*
Strategy							
• *Planning*	Low	High	Med	Med	High	High	High
• *Finance*	Low	Med	Med	Med	Med	High	High
• *Governance*	High	High	Med	High	Med	High	High
Architecture							
• *Business*	Med	High	Med	Med	High	High	High
• *Application*	Low	Med	High	Med	High	High	High
• *Technology*	Low	Low	Low	Low	High	Low	High
Operations							
• *Project Mgmt*	Low	Low	High	Low	Low	High	High
• *Service Mgmt*	Low	Med	High	Med	Med	High	High
• *Quality Mgmt*	Low	Low	High	Low	Med	High	Med

3.1 Scope of EA Maturity Frameworks

Based on the dimensions outlined in section 2.1, we analyze the scope of the selected frameworks (cf.Table 2). For this purpose, we assigned a score to each focus area for every framework, i.e. a high score indicates that the focus area and the corresponding maturity levels are covered in more detail by the assessment types (processes or characteristics). Surprisingly, the strategy dimension generates the highest scores (cf.

Fig.1) which stems from the fact that Governance is best covered by those frameworks. Most notably, COBIT together with ValIT and the IT-CMF possess the best overall scores, although there is only one framework capable of serving an overarching maturity purpose for a value-driven enterprise, namely the IT-CMF for it offers far more detailed maturity models for focus areas even beyond the three dimensions. Needless to say, that this is a high-level inspection to give a general idea of what an overarching maturity framework should be capable of in terms of the selected focus areas.

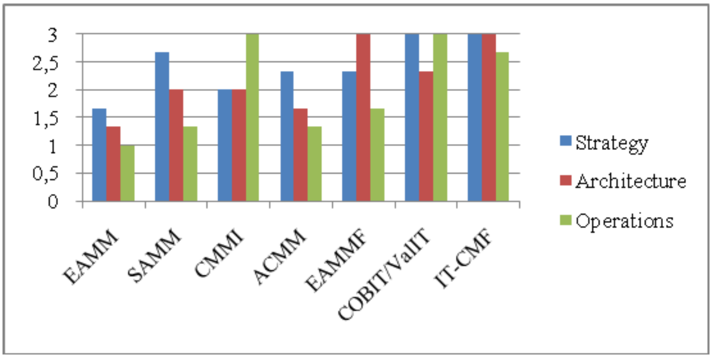


Fig. 1. Scope of EA Maturity frameworks across three dimensions

4 Analyzing EA Maturity Frameworks

Analyzing different maturity models has been done several times in literature. In [20], a comparison between different maturity models for project management is carried out according to 27 variables. The study comprises nine different models and the result is presented in table form where one framework is chosen to be the best suited in terms of selected variables.

Another analysis of different maturity models has been done in [21] where the focus is on IT Governance with the special emphasis on multi-sourcing. For this purpose, seven requirements were identified and used to analyze multi-sourcing maturity models or best practices respectively with a table representing the result.

In order to give more substance to the term EA maturity, we need to address several maturity characteristics that are important for this domain. In other words, for a thorough analysis of EA Maturity frameworks, we need to identify adequate criteria or key characteristics respectively, together with corresponding attributes. These are summarized in Table 3.

Maturity Profile

This profile determines to which target the maturity level is assigned, such as processes, characteristics or focus areas. Another important issue is if there is an assessment aggregation, i.e. are several targets with already assigned maturity levels combined and translated into a bigger picture or overall maturity.

Table 3. Key characteristics and relevant attributes for the framework analysis

Key Characteristic	Relevant Attributes
Maturity Profile	Assessment Targets, Assessment Aggregation, No. of levels, Representation
Assessment Method	Type of Method, Time, Complexity
Maturity Improvement	Type of Directions
Value Proposition	Value/Benefit
EA Domain Coverage	Business Architecture, Application Architecture, Technology Architecture

A maturity profile can be represented in several ways, although a table representation is the most common. Another way to show subject-maturity relationships is to employ a cube-like form [22]. Consequently, subjects are taking up two dimensions whereas the maturity levels the remaining one. Also to be found are mere textual descriptions for every level and the corresponding characteristics.

Many maturity models adopt a five-level profile, although some approaches, such as [5], [23], [13] support more than these five levels in order to allow a finer granularity in terms of improvement steps. This also brings an increased level of complexity and may make the assessment process more difficult and time consuming.

In the course of our examination of maturity profiles, we see that every one of the selected frameworks possesses a level 0. According to general understanding, we do not count it as a full-fledged maturity level. The analysis of the frameworks in terms of the maturity profile is summarized in Table 4.

Table 4. EA Maturity Profiles for selected frameworks

Framework	Assessment Targets	Assessment Aggregation	No. of levels	Representation
IT-CMF	Capability Building Blocks	Critical Processes	5	Table
EAMM	Categories	None	5	Text
SAMM	BITA[2] Criteria	None	5	Text
CMMI	Process Areas, Capabilities	None	5, 3	Table
ACMM	IT Architecture Characteristics	Operating Unit	5	Table
EAMMF/EAAF	Critical Success Attributes	CSA Representations	6	Table/Cube
COBIT/ValIT	Processes	None	5	Table

Assessment Method

The assessment method serves the purpose of assigning the maturity levels to different components. We examine the frameworks in terms of this method and whether it supports a structured way of evaluating the maturity. Furthermore, we inspect the expenditure of time and the difficulty to undertake such assessments. Another aspect of such a method is the involved roles and responsibilities which gives insight to the level of engagement and how far-reaching the scope in terms of organizational involvement is. The analysis of the frameworks in terms of the assessment method is summarized in Table 5.

[2] BITA: Business IT Alignment

Table 5. Assessment Method for selected frameworks

Framework	Type of Method	Time	Complexity
IT-CMF	Questionnaire, unique for every CP	Quick	Medium
EAMM	NASCIO Toolkit	Quick	Low
SAMM	High-level process with 6 steps	Long	High
CMMI	SCAMPI, detailed process, roles and responsibilities	Long	High
ACMM	Scorecard (Questionnaire)	Quick	Low
EAMMF/EAAF	EAAF, practices and artifacts for every level	Moderate	Medium
COBIT/ValIT	None explicitly mentioned	Varies	Medium

Maturity Improvement

Having achieved a certain level of maturity, a company needs to know what is to be done next in order climb up the maturity ladder. Hence, we look at the framework support for directions on how to achieve this progress in terms of maturity.

Notably, it is generally assumed, that every company wants to achieve the highest possible level of maturity, even if it might not be feasible to reach it, especially when considering SMEs. However, this issue needs to be further investigated and is beyond the scope of this paper. Another interesting point for maturity improvements is the involved costs, but since they vary for every level and company, we will not investigate this topic further. Such research was conducted in [24] where an analysis on how to reach level five within *CMMI* is described. The analysis of the frameworks in terms of the maturity improvement is summarized in Table 6.

Table 6. Maturity Improvement for selected frameworks

Framework	Type of Directions
IT-CMF	Available for every CP
EAMM	None
SAMM	Depends on gap analysis, no detailed directions available
CMMI	Detailed in SCAMPI
ACMM	None
EAMMF/EEAF	Given in the EEAF, artifacts for every level
COBIT/ValIT	Depends on goals and metrics defined for every process

Value Proposition

Employing a maturity model assumes that the company plans and operates more efficiently if higher maturity levels are attained. But there is more to the value argument. What kind of value brings the application of the maturity model and how can this be measured? In literature, we can find several contributions that deal with ROI ([24], [25] and references therein) of software process improvement. In terms of software, the software product marks the most important measure of success in any improvement program [24]. The product itself may be developed and maintained by several processes, but improving each process according to the maturity profile does not necessarily mean that the overall product is better in terms of performance or efficiency [26]. This can be broken down to an optimization problem and whether its solution provides a local or global optimization.

We will now examine the selected frameworks in terms of their value proposition. It has to be mentioned that we just examine the primary source of framework information, i.e. the built-in claims of value generation since a thorough study on this matter is beyond the scope of this paper. The analysis of the frameworks in terms of the value proposition is summarized in Table 7.

Table 7. Value Proposition for selected frameworks

Framework	Value/Benefit
IT-CMF	Generate business value
EAMM	Reduced redundancy, provide BITA, improved accuracy during development, reduced system complexity, enhanced information sharing, increased traceability
SAMM	Provide BITA and therefore the value of it
CMMI	Improved performance, reduced costs
ACMM	Provide strategic direction; identify weak spots; enhance overall success of IT Architecture
EAMMF	Identify status quo in a simplified and structured way
COBIT/ValIT	Provide governance, improve efficiency of IT investments

EA Domain Coverage

We already outlined what EA is all about. For analyzing EA Maturity frameworks we need to find viable perspectives or domains respectively. A generally accepted way to organize an EA into domains is to introduce four distinct sub-architectures, namely the Business-, the Data/Information-, the Application-, and the Technology Architecture (e.g. [27]). For the purpose of this work, we just use three domains and include the Data/Information Architecture into the Application Architecture as done in [28]. The reason for this is that data or information respectively can be considered as a passive structure created, managed or analyzed by applications and services. Data without an application is actually useless. The analysis of the frameworks in terms of the EA domain coverage is summarized in Table 8 and is in line with section 3.1.

Table 8. EA Domain Coverage for selected frameworks

Framework	Business Architecture	Application Architecture	Technology Architecture
IT-CMF	High	High	High
EAMM	Medium	Low	Low
SAMM	High	Medium	Low
CMMI	Medium	High	Low
ACMM	Medium	Medium	Low
EAMMF/EAAF	High	High	High
COBIT/ValIT	High	High	Low

5 Conclusion

In this paper, we presented an overview of the current state of the art in EA maturity by analyzing selected frameworks in terms of the maturity profile, the assessment method, the maturity improvement, the value proposition, and the EA domain coverage which we think are key characteristics of such frameworks. Since EA as discipline deals with the enterprise IT holistically, at least on higher level of abstraction, we wanted to elaborate if there is a framework capable of serving as an overarching IT maturity model. As it is the case, we discovered that the *IT-CMF* is able to fulfill this purpose, not only on a high level of abstraction, but also going into detail by means of providing a structured maturity profile and assessment method for each of the containing processes. Finally, this analysis sheds light on the topic of EA maturity and therefore should aid IT managers as well as Enterprise Architects in selecting the appropriate maturity approach best suited for their needs.

6 Outlook and Future Research

When analyzing frameworks, we usually are challenged by the sheer amount of possible ways of adequate criteria and on which level of abstraction such an analysis is undertaken. Therefore, it might be interesting to elaborate a finer granularity of the chosen characteristics, attributes, dimensions and focus areas in order to provide a more thorough analysis. In addition, the selection of frameworks is certainly not exhaustive, so adding further approaches might prove as valuable. As already pointed out in the course of this paper, some topics are beyond the scope of this contribution, such as the value and cost of these frameworks which would require a larger survey across several sectors with a good number of companies participating in such analysis.

References

1. ISO/IEC 42010 - IEEE Std 1471-2000: Systems and software engineering. IEEE (July 2007)
2. de Bruin, T., Freeze, R., Kulkarni, U., Rosemann, M.: Understanding the Main Phases of Developing a Maturity. In: ACIS, Proceedings (2005) Paper 109
3. Lagerström, R., Franke, U., Johnson, P., Ullberg, J.: A Method for Creating Enterprise Architecture Metamodels - Applid to Systems Modifiability Analysis. International Journal of Computer Science and Applications 6(5), 89–120 (2009)
4. Kaisler, S., Armour, F., Valivullah, M.: Enterprise Architecting: Critical Problems. In: Proceedings of the 38th International Conference on System Sciences, Hawaii (2005)
5. van Steenbergen, M., Berg, M., Brinkkemper, S.: A Balanced Approach to Developing the Enterprise Architecture Practice. In: Filipe, J., Cordeiro, J., Cardoso, J. (eds.) Enterprise Information Systems. LNBIP, vol. 12, pp. 240–253. Springer, Heidelberg (2009)
6. SEI Carnegie Mellon University: Capability Maturity Model Integration (CMMI), Version 1.3 (2010)
7. Curley, M.: An IT Value Based Capability Maturity Framework. MIT Sloan CISR VI(2D) (2006)
8. National Association of State Chief Information Officers. Enterprise Architecture Maturity Model, http://www.nascio.org/publications/documents/NASCIO-EAMM.pdf (accessed, December 2003)
9. Luftman, J.: Assessing Business-IT Alignment Maturity. Communications of the Association for Information Systems 4(1), Article 14 (2000)

10. Luftman, J. (ed.): Competing in the Information Age: Align in the Sand, 2nd edn. Oxford University Press, Oxford (2003)
11. SEI Carnegie Mellon University: Standard CMMI Appraisal Method for Process Improvement (SCAMPI), Version 1.3 (2011)
12. Department of Commerce (DoC). ACMM Enterprise Architecture Capability Maturity Model, http://ocio.os.doc.gov/ITPolicyandPrograms/Enterprise_Architecture/PROD01_004935 (accessed 2007)
13. U.S. Government of Accountability Office: Organizational Transformation: A Framework for Assessing and Improving Enterprise Architecture Management (Version 2.0), http://www.gao.gov/products/GAO-10-846G
14. U.S. Office of Management and Budget: Enterprise Architecture Assessment Framework (EAAF), http://www.whitehouse.gov/omb/e-gov/eaaf
15. IT Governance Institute: Control Objectives for Information and related Technology (COBIT) (2007)
16. IT Governance Institute: ValIT, Version 2.0 (2008)
17. Lahrmann, G., Marx, F., Mettler, T., Winter, R., Wortmann, F.: Inductive design of maturity models: Applying the rasch algorithm for design science research. In: Jain, H., Sinha, A.P., Vitharana, P. (eds.) DESRIST 2011. LNCS, vol. 6629, pp. 176–191. Springer, Heidelberg (2011)
18. van der Raadt, B., Hoorn, J.F., van Vliet, H.: Alignment and maturity are siblings in architecture assessment. In: Pastor, Ó., Falcão e Cunha, J. (eds.) CAiSE 2005. LNCS, vol. 3520, pp. 357–371. Springer, Heidelberg (2005)
19. Office of Government Commerce: IT Infrastructure Library. The Stationary Office, London (2007)
20. Khoshgoftar, O.: Comparison of maturity models. In: 2nd IEEE International Conference on Computer Science and Information Technology ICCSIT, pp. 297–301 (2009)
21. Herz, T., Hamel, F., Uebernickel, F., Brenner, W.: Towards a Multisourcing Maturity Model as an Instrument of IT Governance at a Multinational Enterprise. In: Proceedings of the 44th Hawaii International Conference on System Sciences, pp. 1–10 (2011)
22. Welke, R., Hirschheim, R., Schwarz, A.: Service-Oriented Architecture Maturity. IEEE Computer 44(2), 61–67 (2011)
23. van Steenbergen, M., Schipper, J., Bos, R., Brinkkemper, S.: The dynamic architecture maturity matrix: Instrument analysis and refinement. In: Dan, A., Gittler, F., Toumani, F. (eds.) ICSOC/ServiceWave 2009. LNCS, vol. 6275, pp. 48–61. Springer, Heidelberg (2010)
24. McGarry, F., Decker, B.: Attaining Level 5 in CMM Process Maturity. IEEE Software 19(6), 87–96 (2002)
25. van Solingen, R.: Measuring the ROI of Software Process Improvement. IEEE Software 21(3), 32–38 (2004)
26. Walker, A.J.: Enterprise Maturity Models: Have We Lost the Plot? Computer 41(11), 96–98 (2008)
27. The Open Group: The Open Group Architecture Framework (TOGAF) Version 9. (2009)
28. The Open Group: ArchiMate 1.0 Specification (2009)
29. Schekkerman, J.: How to Survive in the Jungle of Enterprise Architecture Frameworks: Creating or Choosing an Enterprise Architecture Framework, 3rd edn. Trafford (2006)
30. Becker, J., Knackstedt, R., Pöppelbuß, J.: Developing Maturity Models for IT Management. Business & Information Systems Engineering 1(3), 213–222 (2009)
31. Lahrmann, G., Marx, F.: Systematization of maturity model extensions. In: Winter, R., Zhao, J.L., Aier, S. (eds.) DESRIST 2010. LNCS, vol. 6105, pp. 522–525. Springer, Heidelberg (2010)

32. van Steenbergen, M., Bos, R., Brinkkemper, S., van de Weerd, I., Bekkers, W.: The design of focus area maturity models. In: Winter, R., Zhao, J.L., Aier, S. (eds.) DESRIST 2010. LNCS, vol. 6105, pp. 317–332. Springer, Heidelberg (2010)
33. Schöenherr, M.: Towards a common terminology in the discipline of enterprise architecture. In: Feuerlicht, G., Lamersdorf, W. (eds.) ICSOC 2008. LNCS, vol. 5472, pp. 400–413. Springer, Heidelberg (2009)
34. Rosemann, M., de Bruin, T.: Towards a Business Process Management Maturity Model. In: ECIS 2005 Proceedings (2005)
35. Jugdev, K., Thomas, J.: Project Management Maturity Models: The Silver Bullets of Competitive Advantage? Project Management Journal 33(4) (2002)
36. U.S. Office of Management and Budget: Federal Enterprise Architecture (FEA), `http://www.whitehouse.gov/omb/e-gov/fea/`

Enterprise Resource Planning (ERP) Systems: Use of Reference Models

Dejan Pajk, Mojca Indihar-Štemberger, and Andrej Kovačič

University of Ljubljana, Faculty of Economics, Kardeljeva pl. 17, 1000 Ljubljana, Slovenia
{dejan.pajk,mojca.stemberger,andrej.kovacic}@ef.uni-lj.si

Abstract. Enterprise resource planning (ERP) systems are used more and more extensively, to automate and enhance business processes. The capabilities of ERP systems can be described by best-practice reference models. The purpose of the article is to present the business process renovation approach with the use of reference models. Although the use of reference models brings many positive effects for business, they are still rarely used in small and medium-sized companies. The reasons for this may be found in the reference models themselves as well as in project implementation methodologies. In the article a reference model based on MS Dynamics NAV is suggested. The reference model is designed using upgraded BPMN notation with additional business objects, which help to describe the models in more detail.

Keywords: ERP solution, reference model, business process renovation, MS Dynamics NAV, BPMN.

1 Introduction

Enterprise resource planning (ERP) systems are used more and more extensively. These systems are generic and the functionality they provide can serve a large variety of enterprises. The implementation of an ERP system involves a process of customising the generic package and aligning it with the specific needs of the enterprise [1]. The decision concerning the purchase of individual modules or their development can only be made on the basis of good knowledge of business needs, which have to be compared with the capabilities of an ERP system [2].

ERP system capabilities are best described by reference models. Reference models are generic conceptual models that formalise recommended practices for a certain domain [3]. It represents one or more pre-engineered and integrated organisational views. The type of reference model could be a business process reference model, or a description of data flows [4]. The use of reference models has many positive effects for business [5, 6]. Despite that, they are still rarely used in small and medium-sized companies. The reasons for this can be found in the reference models as well as in implementation methodologies. For some ERP systems reference models have not been developed yet. Developing models from scratch can be very time- and cost-consuming. Therefore, it is reasonable to reuse existing reference models as a starting point to develop specific conceptual models [7].

J. Grabis and M. Kirikova (Eds.): BIR 2011, LNBIP 90, pp. 178–189, 2011.

The aim of this article is to present the BPR approach with the use of reference models. Our research goals are to:

- review the reference models research area;
- suggest use of reference models;
- upgrade BPMN (Business Process Modelling Notation) with additional information objects that enable the design of reference models on a more detailed level; and
- present the use of reference models.

The structure of the paper is as follows: the first section introduces the business renovation concept and identifies the role of ERP systems. Section two generally describes reference models. Section three suggests the Dynamics NAV purchase reference model based on BPMN. The last section shows how to use reference models in business process renovation. First, an existing (as-is) business process is presented and then, with the use of a reference model, a to-be process is suggested.

2 BPR and ERP Systems

The rapid and constant changes that are very common in today's business environments affect not only business itself, but also its supporting business information systems (IS). As a result, IS require constant change, renovation and adaptation to meet actual business needs.

Business renovation is presented as the highest level of a strategy for managing change that usually cannot be handled by continuous improvement and reengineering methods (BPR) or organisational restructuring [8]. BPR is a reengineering strategy that critically examines current business policies, practices and procedures, rethinks them and then redesigns critical products, processes and services [9].

The implementation of large IS is impossible without first altering business processes; while renovation is essential in order to extract the maximum benefit from IS products. In the past, companies first decided how they wanted to do business and then selected the software package that supported their business processes. They made many modifications to ensure a tight fit. This changed with the introduction of ERP systems that often required business processes to be modified to fit the IS [10]. An ERP system is a business management system that comprises integrated sets of comprehensive software that can be used, when successfully implemented, to manage and integrate all business processes and functions within an organisation. They usually include a set of mature business applications and tools for financial and cost accounting, sales and distribution, management of materials, human resources, production planning and computer integrated manufacturing, supply chain, and customer information [2].

Nowadays many organisations support their processes by purchasing ERP systems. The rate of unsuccessful projects is significantly high. The main reason for this is the underestimation of the complexity of such a project that requires several organisational changes and the involvement of employees. The massive organisational changes involved in ERP implementation result from the shift in business design from a fragmented, functional-based organisational structure to a process-based one [10]. So, it is very important to select an appropriate ERP system. The selection process should be based on a comparison of business needs and the capabilities of an ERP system [2]. Business needs are best described by organisation's strategy and desired business processes. Processes inside an organisation have to be compared with the ERP system's capabilities (Fig. 1) [11].

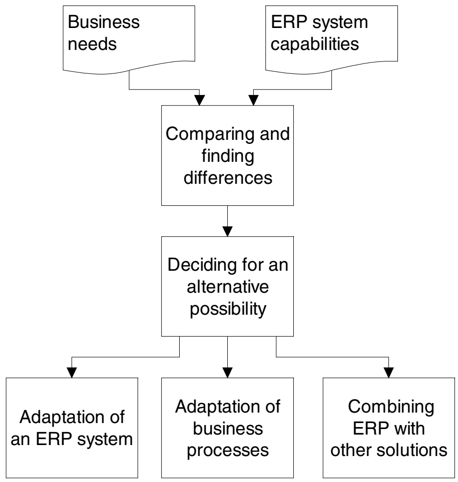


Fig. 1. The alternatives for selecting and implementing an ERP system

Based on the comparison a company can choose between three alternatives. The first is to adapt an ERP system to their business processes. Most ERP systems allow a certain degree of customisations and parameterisation. This alternative can cause high additional costs along with problems in further maintenance and upgrade projects.

The second alternative is the adaption of business processes to an ERP system. Typically, the delivery of best practice applies more usefully to large organisations and especially where there is a required standard, or where the process is a commodity like accounting processes. This is because the procedure of capturing and reporting standardised or commodity content can be readily codified within the ERP software, and then replicated across multiple businesses with the same business requirements.

The disadvantage of this alternative is that an organisation might lose the advantage of a unique and perhaps a better business practice. Best-practice processes are comparable to everyone else in the industry sector and therefore erode competitive advantage. The third alternative is to combine the acquired ERP system, integrated best-of-breed systems and engineered adapted or built applications. This is the best possible alternative for the majority of cases [11]. The comparison therefore requires clear business needs on one side and recognised capabilities of ERP system on the other side. ERP system capabilities can be presented by ERP consultants, user manuals, training materials etc. The best way to formalise the capabilities of an ERP system is by process reference models. They can be used to describe the features of different ERP packages. Based on such a description, it is possible to compare and select an appropriate ERP package for an enterprise.

2.1 Business Process Reference Models

Process design is a key phase in the renovation of a business process. The resulting blueprint is the basis for implementation and execution, as well as monitoring and controlling processes. Ensuring such modelling quality can be very time-consuming. The use of process templates significantly increases the efficiency and effectiveness of the process design phase. The process templates are generally called business process reference models [5].

A reference model encompasses one or more pre-engineered and integrated organisational views. For example, one type of reference model might be a business process reference model, or a depiction of data flows [4]. It is an abstraction to facilitate understanding of the relationships among various objects, and for the development of consistent standards or specifications supporting an integrated environment. A reference model is based on a small number of unifying concepts and may be used as a basis for education and explaining standards to non-specialists [12]. In the literature we can find several other definitions of reference models. In [13] Rosemann defines reference models as generic conceptual models that formalise recommended practices for a certain domain. Fettke and Loos [14] contended that a reference model represents a class of domains.

Reference models have the following characteristics [6, 14, 5, 15]:

- a representation of best practices (providing best practices for conducting business);
- universal applicability (representing a class of domains, not a particular enterprise); and
- reusability (they can be understood as blueprints for developing information systems, they can be structured for easy adaptability to company-specific situations).

Reference models play an increasingly important role in activities such as business engineering [16], information systems development [17] customising of ERP systems [3] and training and research [18]. In order to be able to use reference models, they must be adapted to the requirements of a specific enterprise. Reference models are also called universal models, generic models or model patterns.

Reference models represent the content of various domains. The most important types are the following [5, 14]:

- industry reference models (representing the best practices of a specific industry sector);
- software reference models (these could be traditional applications such as ERP systems, or a reference model representing the sub-process supported by service-oriented architecture (SOA);
- procedural reference models (e.g., a project management reference model); and
- company reference models (representing best practices within a company or a company group).
- Process reference models (Table 1) integrate the well-known concepts of business process reengineering, benchmarking, and process measurement into a cross-functional framework [19].

Table 1. The process reference model concept

Business Process Reengineering	Benchmarking	Best Practices Analysis	Process Reference Model
Capture the "as-is" state of a process and derive the desired "to-be" future state. ⇨		⇨	Capture the "as-is" state of a process and derive the desired "to-be" future state.
	Quantify the operational performance of similar companies and establish internal targets based on "best-in-class" results. ⇨	⇨	Quantify the operational performance of similar companies and establish internal targets based on "best-in-class" results.
		Characterize the management practices and software solutions that result in "best-in-class" performance. ⇨	Characterize the management practices and software solutions that result in "best-in-class" performance.

Certain types of software are designed and developed once, and then replicated many times (e.g. Microsoft Office). Certain software vendors (e.g. SAP, Oracle etc.) have applied this same concept at the enterprise level. They have designed and developed modular standard software solutions that enable business applications to be deployed across the enterprise. The idea is to implement the software with minimum modifications in order to avoid the associated costs and risks [4].

The use of reference, modelling has different economic effects on the modelling process [20, 6, 5]:

- a decrease in costs (reference models can be reused so the development costs of the reference model can be saved);
- a decrease in modelling time (the knowledge contained in the reference model reduces learning and development time, allowing the identification of and a direct focus on critical processes);
- an increase in model quality (reference models are proven solutions and provide better model quality and an awareness of own deficiencies); and
- a decrease in modelling risk (the risk of failures during reference model usage can be reduced because reference models are already validated).

Possible disadvantage of using reference models is that an organisation might lose some advantage of its unique and perhaps better business practices. The best practice represented by reference models is more or less widely used in the industry sector and therefore cannot represent a source of competitive advantage.

3 Purchase Reference Model Design

ERP-specific reference process models describe the main ERP processes on different levels. Depending on the underlying methodology these models include details of the control flow, organisational units, input and output data and business objects. Further, it is usually possible to refer to the relevant part of the online documentation and, at the lowest level, even to the corresponding ERP transaction [21].

The reference model of the ERP solution Microsoft Dynamics NAV (Dynamics NAV) has not yet been published. In this section we therefore suggest a purchase reference model on two levels. The first level presents a general overview of purchase processes. The second, more detailed level explains purchase posting transactions. The design of the general reference model is based on ERP system educational materials on the purchase area [22], while the design of the purchase posting reference model is based on development and consulting experiences in ERP solution implementation projects.

3.1 Purchase Reference Model

The reference model (Fig. 2) represents some ERP system Dynamics NAV purchase processes. It can be used by managers and business consultants to gain a general overview of certain ERP system capabilities. Practical usage of the reference model is expected at lower levels of detail. The structure of the reference model discussed in the following sections is as follows: purchase setup, purchase order management and requisition management.

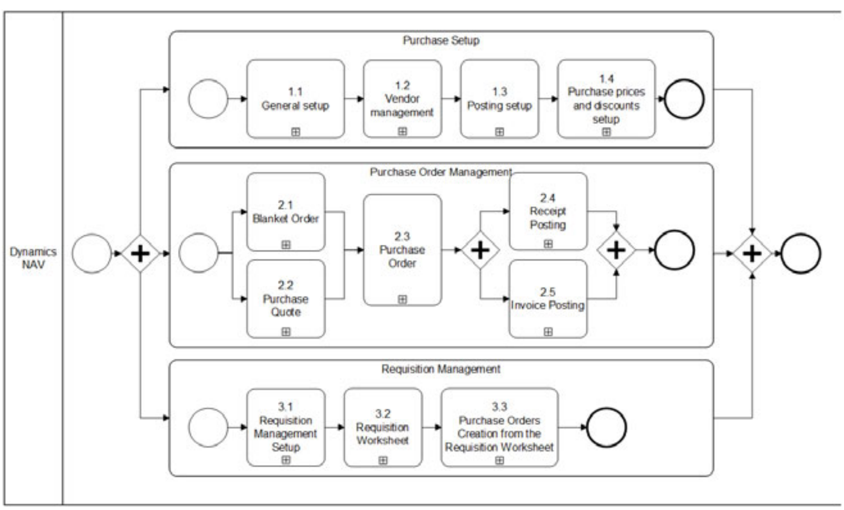


Fig. 2. Purchase reference model (business view)

Purchase setup (1.1) processes elaborate on setup options that define the functionality of the purchase module. Based on established practices, companies must specify how they want the program to support them in managing different aspects of their purchase transactions. These are the general setup options applied to all purchase transactions regardless of which item and vendor are involved. Managing vendor (1.2) information is an important part of managing the total purchases and finances of a company. Basic information (such as name, address and so on) and details (such as credit limit, invoicing, discount and payment terms, currencies, and a list of regularly supplied items) are recorded for each vendor on a vendor card.

Posting setup (1.3) defines the connection between a vendor and the accounts in the general ledger. This is done by assigning a vendor to a posting group for balance sheet and income statement accounts. Companies have the possibility to specify cost and discount information (1.4) for each item on the item card, and are given a functionality that facilitates the task of purchase price management. The program automatically retrieves information about the last direct cost stored on an item card to copy it to the purchase order line for the item in question.

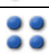
The first step in a workflow involves making a blanket order (2.1) or purchase quote (2.2). A blanket purchase order represents a framework for an agreement between the company and a vendor. Blanket orders are used when the company has committed to buying large quantities of an item that are to be received in several smaller shipments over a certain period of time. A quote can be described as a draft order in which purchasing agents can register the vendor's offer specifying the price, terms of sale, description of items etc. A blanket order and quote can be converted into a purchase order (2.3), which is a cornerstone of purchase management functionality in Microsoft Dynamics NAV. Receipts and invoices are posted from the purchase order.

The requisition management functionality (3.1, 3.2, and 3.3) helps automate the procurement process and enables the purchasing agent to perform basic purchasing activities more efficiently. The requisition worksheet calculates a current and detailed purchase order proposal plan, creates actual purchase orders from order proposal lines, manually handles created purchase order proposal lines, controls the flow of relevant information between the departments concerned, and provides a practical overview of the individual processes involved.

3.2 Purchase Posting Reference Model

This section introduces a more technical view of ERP system dynamics NAV. The purchase posting transaction is presented with corresponding business objects which enable particular activity. In order to design such a reference model, we have upgraded the BPMN with additional ERP system objects (Table 2).

Table 2. ERP solution Dynamics NAV business objects

	Table: Used to describe how data is stored in the database and how it is retrieved
	Form: Used to display data to the user in a familiar and useful way
	Report: Enables users to summarise and print detailed information using the filters and sorting that they choose
	Dataport: Able to export or import table data
	XML port: Related to a dataport; used to import and export data
	Codeunit: Contains program functions

The business object adds additional information to the model. This could be very useful information for ERP system analysts and developers especially in the design and development phases of ERP system implementation. The reference model therefore can represent the basis for technical documentation.

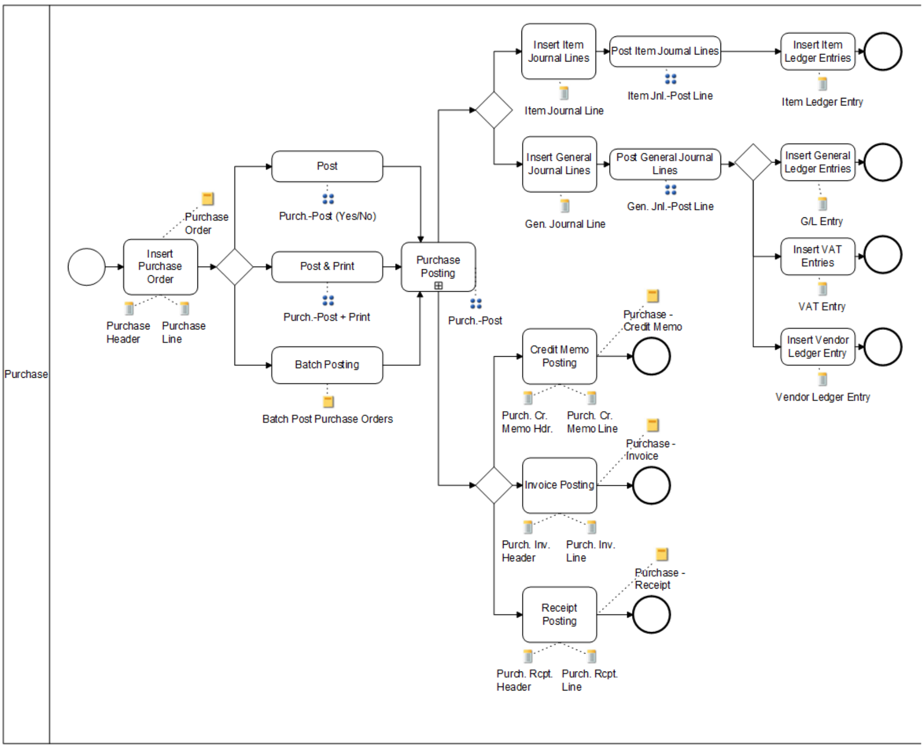


Fig. 3. Purchase posting reference model (technical view)

4 Use of the Purchase Reference Model in Process Renovation

This section shows how to use reference models in a business process renovation. First an existing requisition business process (Fig. 4) in a trading and manufacturing company is presented. The main activities of the selected company are the wholesale of textile goods, products and household services. The sales are focused on the Slovenian market.

With the use of the purchase reference model presented in the previous section, the existing requisition process is renovated and a new to-be process (Fig. 5) is suggested. The to-be process is supported by ERP system Dynamics NAV.

4.1 Purchase Requisitions Process Modelling

The purchase requisitions process (Fig. 4) explains events that can trigger the requisition of a specific product or service. These events can be:

- external, e.g. a paper list from a warehouse or a confirmation or invoice from a supplier;
- a released sales order in the case of a direct delivery where received goods are delivered directly to the customer;
- a confirmed requisition; or
- a requisition plan which has information regarding the optimal inventory level, requisition time periods etc.

Regardless of the source, currently in all cases the purchase order is manually entered into the system by purchasing clerks. The owner of the requisition plan is purchasing manager. The plan is supported by Microsoft Excel.

The presented process involves a lot of manual work, especially with the entering of purchase orders and updating of the purchase plan.

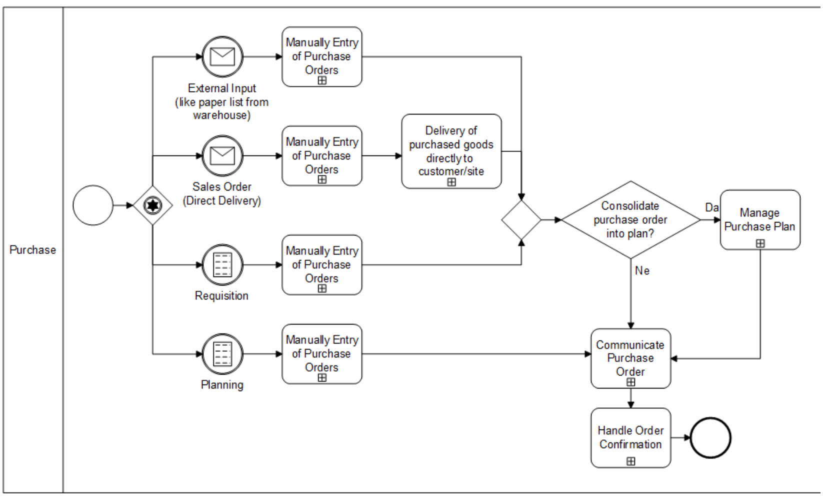


Fig. 4. Purchase requisitions process (as-is)

4.2 Purchase Requisitions Process Renovation

Process renovation is achieved based on a comparison between the purchase reference model of an ERP system Dynamics NAV (Fig. 2), and the purchase requisitions process (Fig. 4). The activities of the new to-be process (Fig. 5) have a corresponding number of reference model sub-processes. This represents a link and explains which component of an ERP system supports the specific activity of the company's process. The result of the comparison shows the degree of fit and how many modifications would be needed on the ERP system side.

A reduction of manual work and many other advantages are achieved with the next process improvements:

- sales order lines, in the case of direct delivery, are now directly transferred to purchase orders (the purchasing clerk does not have to manually enter them again);

- the confirmed requisition is already entered in the system as a purchase quote (a blanket order is converted into a purchase order automatically); and
- a requisition plan based on predefined parameters suggests the necessary requisitions and also automatically creates purchase orders.

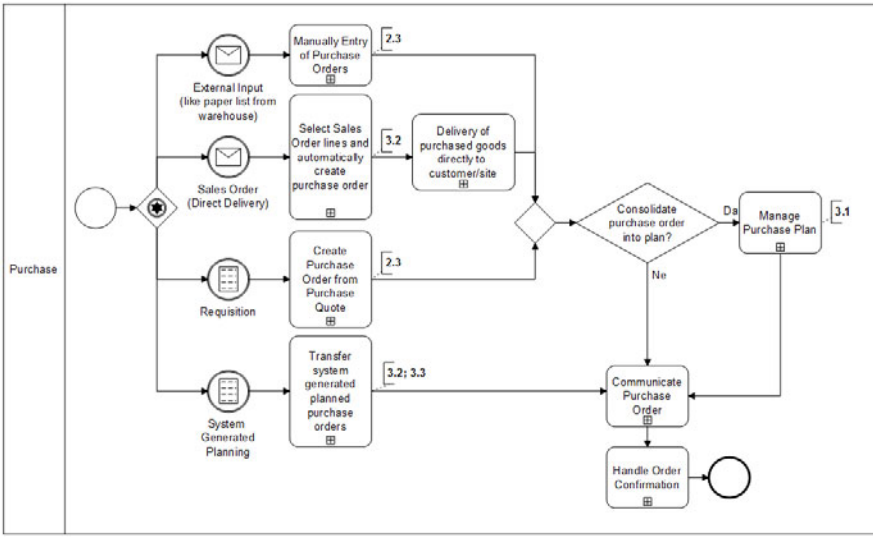


Fig. 5. Purchase requisitions process (to-be)

5 Conclusion

The redesign of business processes and implementation of an IS can represent the best way to face the challenges of today's changing business environment. Information systems require constant change, renovation and adaptation to meet actual business needs.

Conceptual models play an increasingly important role in all phases of the information systems life cycle. For instance, they are used for business engineering, information systems development and the customising of ERP systems. The design of such models is often cost- and time-consuming. The concept of reference modelling has been introduced to overcome these failures and improve the development of enterprise-specific models. Reference models are generic conceptual models which formalise recommended practices for a special domain. They deliver best practice information that can be used many times.

The aim of our article is to present the BPR approach with the use of reference models. In the paper we first present a review of the reference models research area where we suggest the use of reference models in the process of alternative possibilities for selection and implementation of an ERP system.

In the second section we introduced the reference model concept and conclude, that the best way to formalise the capabilities of an ERP system is by process reference models. They can be used to describe the features of different ERP packages. Based on such a description, it is possible to compare and select an appropriate ERP package for an enterprise.

In the third section we suggested the Dynamics NAV purchase reference model based on BPMN. We also upgraded the BPMN with upgrade BPMN (Business Process Modelling Notation) with additional information objects that enable the design of reference models on a more detailed level. The business object adds additional information to the model. This could be very useful information for ERP system analysts and developers especially in the design and development phases of ERP system implementation.

In the last section we presented how to use reference models in business process renovation. First, an existing requisition business process in a trading and manufacturing company was presented. Based on the comparison between the suggested reference model and the as-is process we designed the to-be process and explained the modifications. In our case study we presented many positive affects that reference models have on business. A reduction of manual work and many other advantages are achieved with the process improvements. In the future we should devote more attention to this area, especially in practice.

References

1. Soffer, P., Golany, B., Dori, D.: ERP modeling: a comprehensive approach. Information Systems 28(6), 673–690 (2003)
2. Kovačič, A., Bosilj Vukšić, V.: Management poslovnih procesov: Prenova in informatizacija poslovanja s praktičnimi primeri. GV, Ljubljana (2005)
3. Rosemann, M., van der Aalst, W.M.: A Configurable Reference Modelling Language. Information Systems 32(1), 1–23 (2007)
4. Enterprise Integration Inc., White Paper: What is a reference model? retrieved from, `http://www.eiisolutions.net/resource-center/What%20Is%20a%20Reference%20Model.pdf`
5. Kirchmer, M.: Reference Models to Empower MPE. In: High Performance Through Process Excellence, pp. 85–100 (2009)
6. Fettke, P., Loos, P.: Perspectives on Reference Modeling. In: Reference Modeling for Business Systems Analysis, pp. 1–21. Hershey, London (2007)
7. Becker, J., Beverungen, D.F., Knackstedt, R.: The challenge of conceptual modeling for product–service systems: status-quo and perspectives for reference models and modeling languages. Information Systems and E-Business Management 8(1), 33–66 (2010)
8. Kovacic, A.: Business renovation projects in Slovenia. Business Process Management Journal 7(5), 409–419 (2001), doi:10.1108/14637150110406795
9. Prasad, B.: Hybrid re-engineering strategies for process improvement. Business Process Management Journal 5(2), 178–198 (1999), doi:10.1108/14637159910269728
10. Davenport, T.H.: Putting the enterprise into the enterprise system. Harvard business review 76(4) (1998)
11. Štemberger, M.I., Kovacic, A.: The Role of Business Process Modelling in ERP Implementation Projects. In: Proceedings of the Tenth International Conference on Computer Modeling and Simulation, pp. 260–265. IEEE Computer Society, Los Alamitos (2008)

12. AGIMO, The Australian Government Business Process Interoperability Framework (2007)
13. Rosemann, M.: Application reference models and building blocks for management and control (ERP Systems). Handbook on Enterprise Architecture, 596–616 (2003)
14. Fettke, P., Loos, P.: Classification of reference models: a methodology and its application. Information Systems and e-Business Management 1(1), 35–53 (2003), doi:10.1007/BF02683509
15. Scheer, A.W.: Business Process Engineering Study Edition: Reference Models for Industrial Enterprises, 1st edn. Springer, Heidelberg (1998)
16. Scheer, A.: ARIS: Business Process Modeling, 3rd edn. Springer, Heidelberg (2000)
17. Winter, R.: Formalised conceptual models as a foundation of information systems development. In: ER 1994, vol. 881, pp. 437–455 (1994)
18. Thomas, O.: Understanding the Term Reference Model in Information Systems Research: History, Literature Analysis and Explanation. In: Bussler, C.J., Haller, A. (eds.) BPM 2005 Workshops. LNCS, vol. 3812, pp. 484–496. Springer, Heidelberg (2006), doi:10.1007/11678564_45
19. Supply Chain Council, Supply Chain Operations Reference Model. Version 9.0 (2008)
20. Hilt, B.: Predefined reference models – a »living guide« containing proven process knowledge that anyone can use. IDS Scheer AG: ARIS Expert Paper (2007)
21. Rosemann, M.: Using reference models within the enterprise resource planning lifecycle. Australian Accounting Review 10(22), 19–30 (2000)
22. Microsoft Corporation, Trade in Microsoft Dynamics NAV 5.0 – Training material (2008)

A Universal Model-Based Solution for Describing and Handling Errors

Sergejs Kozlovics

University of Latvia, Faculty of Computing
Raina blvd. 19, LV-1586, Riga, Latvia
Institute of Mathematics and Computer Science, University of Latvia
Raina blvd. 29, LV-1459, Riga, Latvia
sergejs.kozlovics@lumii.lv

Abstract. Before the system is on the way to end users' desktops, developers should not forget about error messages their system might display. That is also true for systems with models inside. Is there a model-based solution for describing and handling errors in a universal way? We say 'Yes', and we show how the meta-meta level of abstraction aids here. Having error types described, particular errors are simply placed into the model repository (when needed), and the error handling mechanism automatically handles and displays them in an appropriate way.

Keywords: error meta-metamodel, model-based error handling, error engine, Transformation-Driven Architecture.

Dear valued user,
You have reached the error page for the error page...
You win!!
—The Best Gmail Error Message, Google blogspot, September 25, 2008

1 Introduction

Error messages are familiar to PC users all over the world. And not only to users: it is the job of developers to decide how the user is informed about one or another error. No, we do not speak about bugs in programs, but about the errors that are expected and of which the developer is aware. "The connection to the network has been lost"; "The file already exists"; "The inheritance loop is detected (in the class diagram)" — these are just some of the examples. The reality is: developers have to deal with such errors. And that is also true when software is model-driven.

Thinking about smooth error messages and structuring them may seem not-so-significant until the system goes to the end user's desktop. Before user's eyes, error messages should be comprehensible. It is not polite to make the end user feel guilty with an obscure technical error message. Also, if there are 100 similar errors (e.g., failed to copy 100 files), it may be reasonable to show only one error

J. Grabis and M. Kirikova (Eds.): BIR 2011, LNBIP 90, pp. 190–203, 2011.

message. Sometimes, it may be useful for developers and advanced users to be able to get the detailed information about the error.

The question arises: how to manage errors in a model-based system easily, while also thinking about the issues just mentioned? We will answer this question by presenting a universal model-based error handling solution. An peculiar property of our solution is that for describing errors it naturally involves the meta-meta level of abstraction.

The idea of a universal error handling mechanism arose during the development of the Transformation-Driven Architecture (TDA) [1]. So, we start by a brief description of TDA (Sect. 2). This allows us to be more concrete when explaining the idea. We start explaining the idea by justifying the explicit usage of the meta-metalevel in our solution (Sect. 3). Then we present the error meta-metamodel (Sect. 4) for describing errors and explain how particular error instances are created and handled (Sect. 5). Although such a universal solution seems to be new in model-driven software development, we mention some related work in Sect. 6 and then conclude.

2 The Essence of the Transformation-Driven Architecture

The Transformation-Driven Architecture (TDA) is an approach for developing model-based software in general and domain-specific tools in particular [1]. TDA has also an implementation. The essence of TDA is depicted in Fig. 1.

Model transformations are in the centre of TDA. They implement the logic of the system. Along with the domain metamodel (metamodels are depicted as ellipses) describing the domain data, there are also several *interface metamodels* describing graphical presentations (e.g., graph diagrams, dialog windows, etc.)

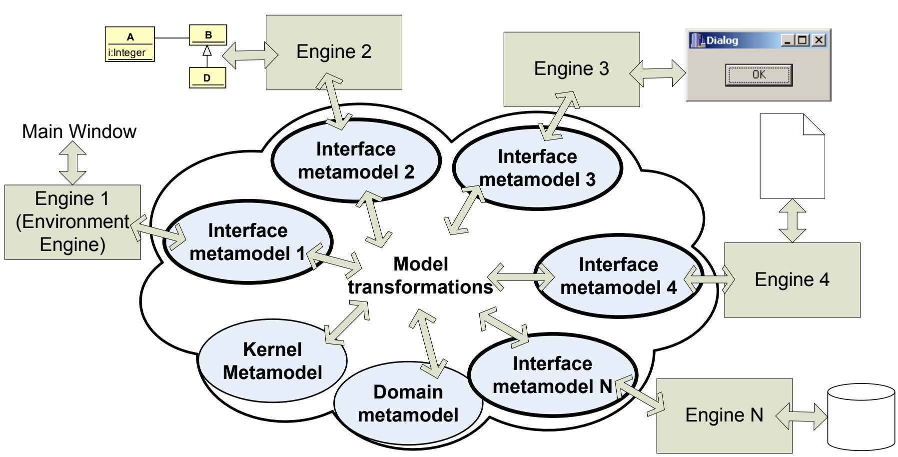


Fig. 1. The Essence of the Transformation-Driven Architecture

and services (e.g., connections to databases, generation of Word documents, etc.) used by model transformations. Modules which provide such presentations and services are called *engines* (depicted as rectangles).

One of the key points that makes TDA different from other approaches refers to interface metamodels of engines. These metamodels are officially parts of the architecture and serve as a boundary between the model-driven world (i.e., model transformations, where the "business logic" resides) and the traditional code world (graphical presentations and services provided by engines). This allows numerous technical implementation details of engines to stay beyond interface metamodels and, thus, be hidden from model transformations. Having only essential information in them, interface metamodels serve as a good documentation for transformation developers. To get more insight on interface metamodels the reader may refer to our publications on Graph Diagram Engine [2] and Dialog Engine Metamodel [3].

Another characteristic feature of TDA is that TDA factors out common functionality used by TDA-based model-driven software. This common functionality is implemented in a universal way by the kernel of TDA. The kernel also has its own metamodel (Kernel Metamodel) describing kernel data and services. Some of the features offered by the TDA kernel are as follows:

– It provides access to model elements stored in a model repository. Engines and transformations access the repository by means of the TDA kernel.
– It provides common communication mechanism between model transformations and engines.
– It provides a universal undo/redo mechanism [4].

Let us explain briefly a communication mechanism offered by the TDA kernel. This will have something in common with the error handling mechanism proposed in this paper (but there will be distinctions, too). Transformations communicate with engines by means of commands, and engines communicate with transformations by means of events. Each command type is a class derived from the *Command* class in Kernel Metamodel. Similarly, events are subclasses of the *Event* class. When a command is given, the kernel calls the required engine. Similarly, when an events occurs, the kernel calls the corresponding transformation assigned to this event. Events and commands may have parameters specified in attributes or in the form of links to other objects.

TDA have proven its success in developing tools for business process management [5], editing UML diagrams [6], querying semantic web data [7], and editing OWL ontologies [8]. Actually, these tools have been created by means of a graphical tool-building platform GRAF (formerly, GrTP) [9,10,11], which itself uses principles of TDA.

The following engines have been already implemented in TDA:

– Graph Diagram Engine (for visualizing diagrams; it uses advanced layout algorithms) [2];
– Dialog Engine (for displaying dialog windows) [3];

– Multi-user Engine (for sharing diagrams and files between several users using SVN);
– Word Engine (for creating Word documents from templates);
– Database Engine (for connecting to databases using ADO).

All these engines may be re-used in different tools built upon TDA. Also, other (e.g., third-party) engines may be plugged in for particular tools, when needed.

During the initial development of TDA, we did not think about error handling much. However, when developing the multi-user engine for sharing diagrams between several users, we had to deal with errors like "Lock failed" or "Commit failed". Of course, the user has to be informed about such errors. And, if several diagrams are committed, the user should know which commits have succeeded and which failed without pressing 100 times the "OK" button. That impelled us to think on how to specify errors in TDA and how to construct an error engine for handling them in a universal way to be able to re-use this engine in different tools. Actually, the error engine turned out to be different from ordinary engines since it needs to work with error metamodels, which are instances of a more abstract meta-metamodel. The following section explains how the need for the meta-meta level arises and describes the basic principles of the error engine.

3 Error Handling and the Meta-meta Level of Abstraction

Modern languages and environments supporting object-oriented programming (OOP) have the notion of *exception*. Usually, all exception types are defined as classes, which form some exception hierarchy. Fig. 2 shows a fragment of Java exception hierarchy [12] in the form of a metamodel.

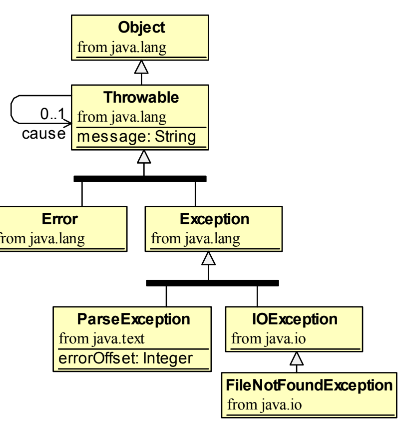


Fig. 2. A fragment of the Java exception hierarchy

As we can see, exceptions are subclasses of the *java.lang.Exception* class.[1] Each exception class may have attributes describing details of the exception (see attribute *errorOffset* in class *ParseException*). The *cause* property allows exceptions to be chained, since one exception may cause another exception to be thrown.

The .NET Framework [13] also has its own exception hierarchy. Exceptions are derived from the *System.Exception* class, which has the *InnerException* property similar to the *cause* property in Java.

Exception classes naturally correspond to the meta-level (Level 2), in which metamodels are described. Particular occurrences of exceptions are instances (objects) of those classes, which correspond to Level 1 of abstraction.

To be able to define error types in TDA, we introduce the *Error* class in Kernel Metamodel — a root class for the error hierarchy. The *Error* class is similar to the *Throwable* class in Java. The *Error* class can be extended in interface metamodels of engines by defining types of errors the engines can produce. Thus, defining new types of errors is similar to defining new types of events and commands in TDA.

However, there is also a difference between events/commands and errors. Commands are processed by engines, and there is a finite number of command types the particular engine has to be aware of. The same is with events. When some transformation is assigned to handle certain events, the number of events the transformation has to be aware of is also finite.

But errors are processed by the error engine, which cannot foresee which engines will be plugged in, and which errors types they will define. Thus, the error engine has to deal with a potentially unbounded number of error types.

But still, the error engine knows something. It knows that the errors form a hierarchy. Also, error messages for errors of the same class should be similar (possibly, with some variations). Thus, some class-level attribute for the error message would come in handy. Since the notion of class hierarchy and the notion of class-level attribute are from the meta-meta level (Level 3), we see that what the error engine could know in advance is not the error metamodel, but the error meta-metamodel.

Fig. 3 depicts, how the error engine embeds into TDA. Error Engine Metamodel consists of the whole TDA error hierarchy (class *Error* and all its descendants) as well as of the two events, which will be introduced later. Error Engine Metamodel is not simply yet another interface metamodel, since it contains error classes from several interface metamodels, and we do not know in advance how many error classes there will be in the hierarchy. Also, to deal with Error Engine Metamodel, the error engine needs to know Error Meta-metamodel, which describes error types from Error Engine Metamodel. Then, given a particular error instance (Level 1, not shown in Fig. 3) and Error Meta-metamodel (Level 3), the error engine can obtain the required information from the error type residing in Error Engine Metamodel (Level 2) and display the error message.

[1] The class *Exception* is derived from *Throwable*, because there exists another subclass derived from *Throwable* — the class *Error*, and Java differentiates exceptions from errors.

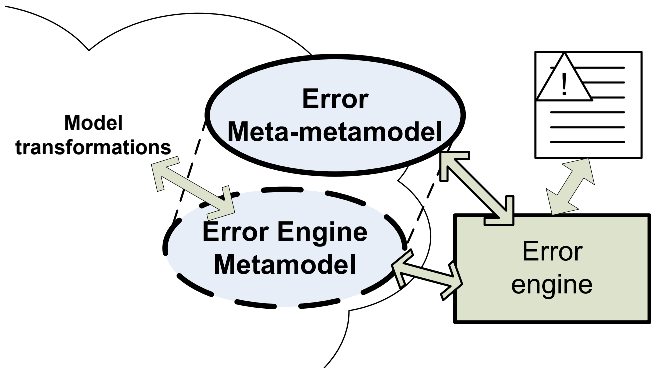


Fig. 3. Adding the error engine to TDA

Error Meta-metamodel contains some metaclasses (types for error types), which define certain class-level properties. These properties will be used by the error engine to decide how to display errors of each particular type. Since errors of the same type have to be displayed in the same way, it is reasonable to specify these properties only once for the given error type (that is why they are class-level properties). The next section will present Error Meta-metamodel.

4 Describing Errors

Error Meta-metamodel for describing error types is depicted in Fig. 4. Classes with shadows are metaclasses — their instances are Level 2 classes. The *Class* metaclass is from EMOF (Essential MOF, [14]), a standard meta-metamodel from OMG. The *Class* metaclass is shown to emphasize that its descendants *ErrorClass*, *WaitingErrorClass*, *NonWaitingErrorClass*, and *AccumulatingErrorClass* are also metaclasses, which define additional properties for error types[2].

Error types (error classes) will be instances of metaclasses *WaitingErrorClass*, *NonWaitingErrorClass*, and *AccumulatingErrorClass*, which all inherit from the common abstract super-metaclass *ErrorClass*. Error types will also be direct or indirect subclasses of the common superclass *Error*, which is an ordinary class in Level 2.

Since error types are instances of metaclasses, the inheritance relationship (corresponding to the EMOF relation *superClass* in the *Class* metaclass) between error types is possible. It can be used to form error hierarchies like in Fig. 5(a).

The *parent-child* relationship of the *Error* class is used to navigate to deeper errors, which caused the particular error object to be created, see Fig. 5(b). The

[2] Technically, the *Class* meta-meta class may be left untouched. For example, the underlying model repository such as JR [15] may support creating metaclasses. Also, class-level properties may be stored as annotations, or encoded in technical Level 1 instances.

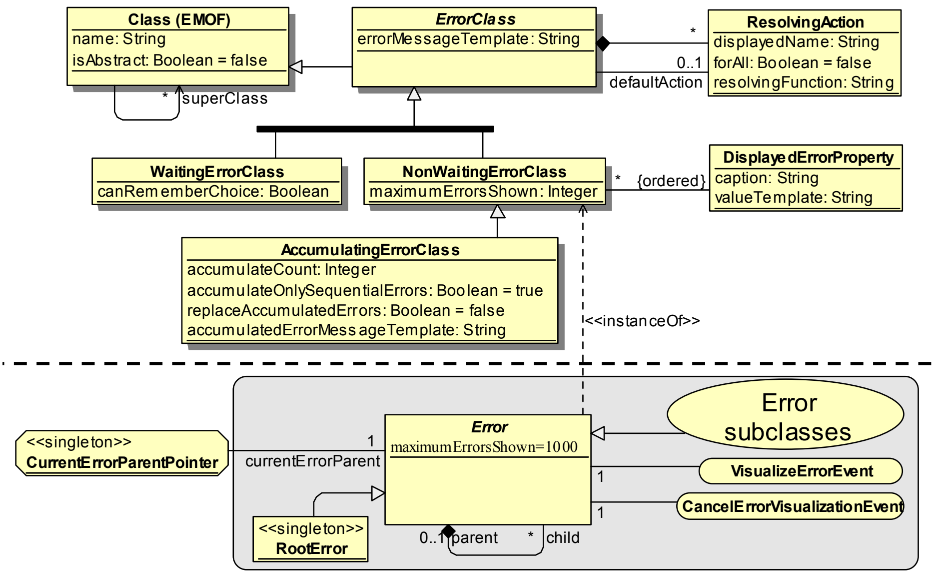


Fig. 4. Error Meta-metamodel (on top of the bold dashed line) and addtional meta-level classes *CurrentErrorParentPointer*, *Error* and *RootError*. Error Engine Meta-model is surrounded by the rounded rectangle.

parent-child relationship is similar to the Java *cause* property and to the .NET *InnerException* property, but we allow not simply chains of "inner" errors like in .NET and Java, but also trees. We allow this, because several consequtive errors, not just one, could lead to the "outer" error (see Fig. 5(c)).

The following subsections will explain more in detail classes found in Error Meta-metamodel.

4.1 *ErrorClass*

The *ErrorClass* class defines common properties for all error types.

The class-level attribute *errorMessageTemplate* is intended for storing the error message common to all errors of the given type. For example, there may be an error class called *FileNotFoundError* having the *errorMessageTemplate* value equal to:

"`File not found!`" .

Sometimes it is reasonable to include some details of a particular error instance, e.g., a particular file name:

"`File document.txt not found!`" .

Such variations could be allowed by making the *errorMessageTemplate* attribute computable (e.g., by assigning a getter function). However, this is not convenient,

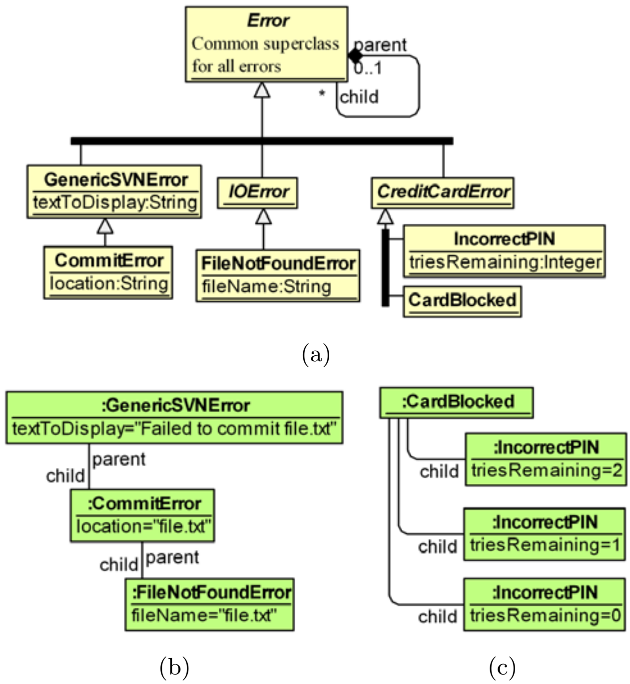


Fig. 5. (a) A sample error metamodel (an instance of Error Meta-metamodel). (b) Sample error instances and the *parent-child* links. (c) Sample error instances, where the *parent-child* links form a tree.

since defining an error type would mean defining also a function. A parameterized string could do the job as well. For example, we could write:

“`File ${fileName} not found.`” ,

where `${fileName}` denotes the value of the attribute *fileName* of a particular *FileNotFoundError* instance (this attribute must exist). And, if computable attributes are supported, the *fileName* value can also be calculated, when needed.

Another way of introducing variations in error messages is by means of OCL-like expressions, e.g.,

“`File ${self.fileName} not found`
`on the remote computer ${self.computer.name}.`” .

(We assume that we can navigate from the error instance by the “computer” role and than take the value of the *name* attribute.) For convenience, `self` pointer could be omitted, so we could write `${fileName}` instead of `${self.fileName}`. OCL-like expressions do not deny computable attributes, but provide additional flexibility in defining error messages.

Sometimes errors are also requests to the user for an action. For example, when copying several files, some file may already exist, and the user should choose between options like 'Yes', 'Yes for all', 'No', 'No for all', etc. The proposed Error Meta-metamodel allows specifying such options as *ResolvingAction*s. When the user selects an option, the corresponding *resolvingFunction* is invoked (this function may be a model transformation or some other function; it must take an *Error* instance as an argument). The *forAll* attribute denotes that the given resolving action has to be invoked for all errors of the same type. The *displayedName* attribute contains the name of the operation shown to the user (e.g., "Yes for all").

In certain cases it is not obligatory to wait for the user to choose the action, e.g., the system may continue copying other files. However, the error object should contain all necessary information (the context) in order the chosen action could be performed later.

Depending on whether the user interaction is required before the processing can be continued we divide all error types into two disjoin sets — waiting error types and non-waiting error types. The following two subsections describe metaclasses *WaitingErrorClass* and *NonWaitingErrorClass* corresponding to the two sets of error types just mentioned.

4.2 *WaitingErrorClass*

Waiting errors require user interaction before the processing can be continued. The expected behaviour[3] of the error engine is to display a modal window with possible choices (resolving actions) how to resolve the error. The *canRememberChoice* attribute denotes whether the error engine has to provide an option (e.g., a checkbox) for applying the same choice for further errors of the same type.

4.3 *NonWaitingErrorClass*

When a non-waiting error occurs, the processing can continue without waiting for the user interaction. The expected behaviour of the error engine on such an error is to append it to some error list shown to the user (compiler warning and errors are usually shown in this way). The *maximumErrorsShown* attribute may be used to limit the number of error messages shown in the list: the older messages will automatically disappear from the screen, when the number of errors exceeds the value of *maximumErrorsShown*.

If a non-waiting error has resolving actions, the error engine displays the corresponding options (e.g., some buttons near the error message), but does not wait for the user to click on them. When the user clicks on some option, the corresponding resolving function is placed into the queue — it will be executed after the main processing finishes.

The error list mentioned above actually can be a table. In compiler messages file names and line numbers containing errors are usually shown in separate columns. Such columns are instances of the *DisplayedErrorProperty* metaclass.

[3] Although we mention the expected behaviour, there may be different variations of the error engine, and certain features may be implemented there in some other manner.

Each such column has a *caption* and a *valueTemplate*. Each value template is a string, which describes the values for the corresponding column. It can also contain OCL-like expressions, e.g., `${this.source.fileName}` and `${this.token.lineNumber}`.

Also, when too many similar errors occur, some cumulative error may be shown to the user (in addition to existing errors or not). For example, a compiler may summarize how many errors and warnings were there in code, or a function for copying files may show only one error message about 100 files instead of 100 messages, each about a single file. For dealing with such error types, we have introduced the *AccumulatingErrorClass* class, which is explained in the following subsection.

4.4 *AccumulatingErrorClass*

For errors, which are allowed to be accumulated, there is the metaclass *AccumulatingErrorClass*. The value of the *accumulateCount* attribute specifies the threshold for the number of errors, when errors are to be grouped. Until the threshold is reached, accumulating errors are shown as non-waiting errors. The *accumulateOnlySequentialErrors* value specifies whether the threshold can be reached only by sequential errors of the given kind, or errors of other kinds may intervene without restarting the counter. When the threshold has been reached, the previous *accumulateCount*−1 not-accumulated error messages are replaced or appended (depending on the *replaceAccumulatedErrors* value) by the accumulated error message specified as a value of the *accumulatedErrorMessageTemplate* attribute.

Example 1. The value of *replaceAccumulatedErrors* for compiler errors may be set to `false`, and the value for *accumulatedErrorMessageTemplate* may be: “`Too many syntax errors.`”.

If several kinds of errors need to be accumulated, a common supertype may be introduced.

Example 2. Having two error types *FileCopyError* and *FolderCopyError*, we can introduce the common superclass *CopyError*. The *errorMessageTemplate* for *FileCopyError* and *FolderCopyError* may be, respectively, “`Could not copy the file ${fileName}.`” and “`Could not copy the folder ${folderName}.`”, while the *accumulatedErrorMessageTemplate* for the superclass *CopyError* may be: “`Could not copy ${FileCopyError.allInstances->size()} files and ${FolderCopyError.allInstances->size()} folders.`” .

5 Handling Errors

In the previous section we presented Error Meta-metamodel for describing errors. In this section we explain how error instances are created by model transformations and engines, and how they are handled by the error engine.

The Transformation-Driven Architecture has an undo/redo mechanism managed by the TDA kernel [4]. When undo/redo is invoked, a bundle of actions is being undone/redone at once. Each such bundle of actions is a single logical action from the user's point of view. When the user starts a new logical action, he/she usually is not interested in errors from previous logical actions. Thus, we assume that only errors within one logical action have to be managed and stored in the model repository according to Error Engine Metamodel[4].

Since the kernel knows when a new bundle of actions (a new logical action) starts, it performs some cleanup by deleting previous error objects from the model repository. Before each new bundle of actions the kernel also re-creates a singleton instance of the *RootError* class (see Fig. 4) and sets the current error parent pointer (a singleton instance of the *CurrentErrorParentPointer* class) to the root error object.

Then transformations and engines perform their tasks and communicate by means of events and commands. During their work, errors may occur. To report an error an engine/transformation simply creates an instance of the corresponding error type and attaches it to the current error parent.

The error engine will display only errors attached to the root error object, i.e., nested errors will not be displayed[5].

Assume that an engine/transformation needs to call some subroutine, which can also produce errors and store them in the repository. If those errors are too technical and should not be shown to the user, the engine/transformation may create a stub parent error object and move the current error parent pointer to this stub object. From this moment errors created by the subroutine will be attached to the stub object. Since these errors are not directly connected to the root error (e.g., *IncorrectPin* errors from Fig. 5(c)), they will not be shown. When the subroutine finishes, the engine/transformation may replace the stub object with some more suitable error object (depending on the nested errors produced by the subroutine), keeping the nested error hierarchy, when needed (e.g., for providing additional details). Then, the current parent pointer should be moved back to the previous parent.

The error engine, in its turn, is triggered each time a new error object is created (the kernel allows triggers to be defined on certain actions within the model repository). The error engine also follows the current parent pointer link to determine which errors have to be displayed and which not. When an error has to be displayed, the error engine retrieves the corresponding class-level information stored according to the Error Meta-metamodel, and automatically displays the error (e.g., via a modal message box, or by appending it to the list of errors).

For certain errors it is not sufficient to simply show error messages. Some error visualization may be expected, when the user clicks on an error message. For example, if the error is in a graph-like diagram, the corresponding graphical

[4] The error engine may still display the whole history of error messages, but other engines and transformations work only with errors within the current logical action.

[5] Actually, the error engine may provide access to these nested errors for developers and advanced users, but for simplicity we assume that nested errors are not shown.

element may be displayed in a different color. If the error is a syntax error in some code, the caret may go to the corresponding line.

We use the standard TDA event mechanism to inform transformations, when the errors should be visualized. We introduce two event classes: *VisualizeErrorEvent* and *CancelErrorVisualizationEvent* (see Fig. 4). When some error has to be visualized, a *VisualizeErrorEvent* instance is created by the error engine and linked to the corresponding error instance. To handle this event, the TDA kernel will call the assigned transformation, which will take care of visualizing the error. The event *CancelErrorVisualizationEvent* is created and handled in a similar way when there is no more need to have the error visualized.

6 Related Work

Windows API [16] is an excellent example of detecting and displaying errors in a procedural world. Windows functions usually do not display error messages, but in case of an error some error-indicating value (like zero, `NULL`, `-1`, `false`, etc.) is returned. Such a value only indicates that some error has occurred. More details (e.g., the kind of the error) can be obtained by calling the *GetLastError* function, which returns the corresponding integer value. The programmer may decide, whether to format and show the error message depending on that integer value, or to handle the error in some other way, perhaps, also returning some error-indicating value. Returning and checking such values is very fast — each of these values usually occupies a single machine word. However, a single integer value may be insufficient to provide all the details about the error. Certain functions can be introduced for providing additional details, but then this approach becomes not so elegant since the programmer has to know which of these additional functions to call for each error type. In contrast, our model-driven approach is not so fast: instead of dealing with a machine word, we have to create an error object in the model repository. However, the performance price we pay gives us the following benefits:

- The detailed information about the error can be specified in properties of the error object. Thus, no additional functions are needed for that.
- The programmer does not have to format the message: each error class has a message template, which can be automatically transformed to a real error message by the error engine.

In the object-oriented world, error handing is usually based on exception hierarchy. Exceptions are classes, which may have properties for encoding detailed information about errors. Our solution also uses similar principles — error types correspond to classes, which form error hierarchies. Besides, our Error Metametamodel allows the toolsmith to specify possible actions for resolving errors (such as "Yes", "No", "Yes for all", etc.) and properties, which should be displayed in a tabular form. The universal error engine automatically formats and displays the corresponding error message freeing the programmer from that job. In addition, our solution allows several errors of the same kind to be grouped together.

There are also some model-based solutions for managing errors. For example, the work of Brambilla et al. [17] introduces two classes for handling errors: the *ExceptionType* class with the *name* attribute and the *ExceptionInstance* class with three specific attributes. Each exception instance is linked to exactly one exception type.

Another example can be found in the presentation metamodel of the MOLA tool based on Metaclipse [18]. There exists a *CompilerMessage* class with two attributes — one for the error message, and another for the error type (warning or error). There is also a transformation that is used to visualize errors graphically.

Although both mentioned model-based approaches are excellent for particular purposes, they are not universal. The universal (to a wide extent) approach proposed in this paper seems to be new in model-driven development.

7 Conclusion

In this paper, we have presented a universal model-based solution for describing errors (by means of Error Meta-metamodel) and for handling and displaying them (by means of the error engine). We find that the meta-meta level is useful, when a universal solution is being searched.

Currently, we are working on a new version of TDA, which will include the error engine described in this paper. The upcoming version of TDA will provide support for several meta-levels (when using an appropriate repository such as JR [15]) and triggers, which are essential to the proposed error mechanism. We hope the ideas presented in this paper will also find their places in other model-driven solutions.

Acknowledgements. The author gratefully acknowledges the help of (alphabetically) J. Barzdins, A. Kalnins, R. Liepins, E. Rencis and A. Sostaks.

This work has been supported by the European Social Fund within the project "Support for Doctoral Studies at University of Latvia".

References

1. Barzdins, J., Kozlovics, S., Rencis, E.: The Transformation-Driven Architecture. In: Proceedings of DSM 2008 Workshop of OOPSLA 2008, Nashville, Tennessee, USA, pp. 60–63 (2008)
2. Barzdins, J., Cerans, K., Kozlovics, S., Rencis, E., Zarins, A.: A Graph Diagram Engine for the Transformation-Driven Architecture. In: Proceedings of MDDAUI 2009 Workshop of International Conference on Intelligent User Interfaces 2009, Sanibel Island, Florida, USA, pp. 29–32 (2009)
3. Kozlovics, S.: A Dialog Engine Metamodel for the Transformation-Driven Architecture. In: Scientific Papers, University of Latvia vol. 756, pp.151–170 (2010)
4. Kozlovics, S., Rencis, E., Rikacovs, S., Cerans, K.: A kernel-level UNDO/REDO mechanism for the Transformation-Driven Architecture. In: Databases and Information Systems VI - Selected Papers from the Ninth International Baltic Conference, DB&IS 2010 (2010)

5. Barzdins, J., Cerans, K., Kalnins, A., Grasmanis, M., Kozlovics, S., Lace, L., Liepins, R., Rencis, E., Sprogis, A., Zarins, A.: Domain specific languages for business process management: a case study. In: Proceedings of DSM 2009 Workshop of OOPSLA 2009, Florida, USA, pp. 34–40 (2009)
6. The GradeTwo tool, http://gradetwo.lumii.lv/
7. Barzdins, G., Liepins, E., Veilande, M., Zviedris, M.: Ontology enabled graphical database query tool for end-users. In: Databases and Information Systems V - Selected Papers from the Eighth International Baltic Conference, DB&IS 2008 (2008)
8. Barzdins, J., Barzdins, G., Cerans, K., Liepins, R., Sprogis, A.: OWLGrEd: a UML style graphical notation and editor for OWL 2. In: Proceedings of OWLED 2010 (2010)
9. Sprogis, A., Liepins, R., Barzdins, J., Cerans, K., Kozlovics, S., Lace, L., Rencis, E., Zarins, A.: GRAF: a graphical tool building framework. In: Gerard, S. (ed.) Proceedings of the Tools and Consultancy Track of ECMFA 2010, CEA LIST (2010)
10. Barzdins, J., Cerans, K., Kozlovics, S., Lace, L., Liepins, R., Rencis, E., Sprogis, A., Zarins, A.: MDE-based Graphical Tool Building Framework. In: Scientific Papers, University of Latvia (2010)
11. Barzdins, J., Zarins, A., Cerans, K., Kalnins, A., Rencis, E., Lace, L., Liepins, R., Sprogis, A.: GrTP: Transformation based graphical tool building platform. In: Proceedings of MDDAUI 2007 Workshop of MODELS 2007, Nashville, Tennessee, USA (2007)
12. Java^{TM} Platform, Standard Edition 6 API Specification, http://java.sun.com/javase/6/docs/api/
13. .NET Framework Developer Center, http://msdn.microsoft.com/netframework
14. Meta-Object Facility (MOF), http://www.omg.org/mof
15. Opmanis, M., Cerans, K.: Multilevel data repository for ontological and meta-modeling. In: Databases and Information Systems VI - Selected Papers from the Ninth International Baltic Conference, DB&IS 2010 (2011)
16. Richter, J.: Programming Applications for Microsoft Windows, 4th edn. Microsoft Press, Redmond (1999)
17. Brambilla, M., Ceri, S., Comai, S., Tziviskou, C.: Exception handling in workflow-driven web applications. In: Proceedings of International Conference on World Wide Web, pp. 170–179. ACM Press, New York (2005)
18. Kalnins, A., Vilitis, O., Celms, E., Kalnina, E., Sostaks, A., Barzdins, J.: Building tools by model transformations in Eclipse. In: Proceedings of DSM 2007 Workshop of OOPSLA 2007, pp. 194–207. Jyvaskyla University Printing House, Montreal, Canada (2007)

Validating Organizational Knowledge Patterns: Case Study from Information Demand Modeling

Kurt Sandkuhl

The University of Rostock, Institute of Computer Science
Chair Business Information Systems, Ulmenstr. 69, 18051 Rostock, Germany
Kurt.Sandkuhl@uni-rostock.de
Jönköping University, School of Engineering
Box 1026, 55 111 Jönköping, Sweden

Abstract. Among the many concepts and approaches for achieving reusability of enterprise knowledge, patterns of organizational knowledge have been proposed. The purpose of such patterns is to contribute to more efficient operations, avoid repeating costly mistakes or guide continuous process improvements by facilitating systematic reuse of knowledge. The focus of this paper is on developing and validating organizational knowledge patterns in an enterprise context. The paper proposes an iterative development process for organizational knowledge patterns with alternating development and validation phases. The validation phases include a combination of internal and external validation in both theory and practice. The main contributions of this paper are (1) clarification of the term organizational knowledge pattern, (2) an approach for developing and validating such patterns for an industrial context, and (3) experiences from using this approach.

Keywords: organizational knowledge pattern, information demand, validation.

1 Introduction

The context for work presented in this paper is the field of enterprise modeling, which aims at capturing organizational knowledge about processes, products, organization structures and resources in well-defined models contributing to solving enterprise problems [1]. Such models have also been proposed for making enterprise knowledge reusable, for example to visualize best practices how to perform a certain activity or proven ways how to capture product knowledge. In many industrial areas, the intention is to achieve more efficient operations, avoid repeating costly mistakes or guide continuous process improvements by systematically reusing knowledge. Among the many concepts and approaches for achieving reusability of enterprise knowledge, patterns of organizational knowledge have been proposed (see section 2) and – to take one example for organizational knowledge patterns - the concept of information demand patterns has been developed. Information demand patterns focus on information flow in organizations and present typical solutions for demand-oriented information supply, which can be applied in different organizational contexts.

J. Grabis and M. Kirikova (Eds.): BIR 2011, LNBIP 90, pp. 204–216, 2011.

The focus of this paper is on developing and validating organizational knowledge patterns in an industrial context. We argue that a combination of scientifically grounded research methods and application-oriented validation "in use" is crucial for patterns with practical usability and relevance for research. The paper proposes an iterative development process for organizational knowledge patterns with alternating development and validation phases. The validation phases include a combination of internal and external validation in both theory and practice. The main contributions of this paper are (1) clarification of the term organizational knowledge pattern, (2) an approach for developing and validating such patterns for an industrial context, and (3) experiences from using this approach for information demand patterns.

The remaining part of this paper is structured as follows: section 2 describes the concept of organizational knowledge patterns and briefly discusses related work. Section 3 presents our approach for developing and validating organizational knowledge patterns for industrial use. Section 4 illustrates this approach by taking information demand patterns as example. Section 5 investigates whether the approach is suitable for other types of organizational knowledge patterns. This leads to conclusions regarding limits and potential of the approach. Section 6 summarizes the paper and presents an outlook on future work.

2 Knowledge Patterns

The notion of organizational knowledge patterns is closely related to knowledge patterns in computer science and related work in this field. This section will introduce and define the term of organizational knowledge pattern and discuss related work.

2.1 Knowledge Patterns

For more than a decade, patterns have been popular in computer science and were introduced for numerous areas, like software design, information modeling or business processes. Although there is no generally accepted definition of the term pattern, most publications in the field get some inspiration from Christopher Alexander's definition: "Each pattern describes a problem which occurs over and over again in our environment, and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice" [2].

The term knowledge pattern has been explicitly defined by Clark, Thomson and Porter in the context of knowledge representation [3]. They define "a pattern as a first-order theory whose axioms are not part of the target knowledge-base, but can be incorporated via a renaming of the non-logical symbols" [3, p.6]. The intention is to help construct formal ontologies by explicitly representing recurring patterns of knowledge, so called theory schemata, and by mapping these patterns on domain-specific concepts. This work is highly relevant for our work since at its heart ontology construction and organizational knowledge modelling both are modelling endeavors.

Staab, Erdmann and Maedche [4] investigated the use of so called "semantic patterns" for enabling reuse across languages when engineering machine-processable knowledge. Semantic patterns consist in this approach of one description of the core elements independently from the actual implementation and for each target language a description that allows for translating the core elements into the target language. The structure of the informal description consists of eight elements, which remind a bit of design patterns (e.g. name, intent, motivation, structure, etc.); the translation into a language includes translation mapping, samples, applicability and comments. Compared to knowledge patterns, semantic patterns try to separate engineering knowledge from language-specific implementations and not theories from domains they are applied in.

Knowledge formalization patterns have been proposed by Puppe as rather simple templates proven in practice for the (mass) formalization of knowledge [5]. Puppe puts a lot of emphasis on proven problem solving methods, which uncover implicit knowledge from experts. Knowledge formalization patterns consist of well-defined problem solving methods, graphical notation, and simple-to-understand mental model.

It should be noted that there are many more developments in the field of reusable organisational knowledge which are not explicitly called knowledge patterns or semantic patterns. Other developments in this area are for example work flow patterns proposed by van der Aalst and colleagues [6] or patterns for groupware [7].

2.2 Organisational Knowledge Patterns

In order to define the term organisational knowledge pattern, we propose to extend Clark's knowledge patterns by emphasizing more explicitly the focus on characteristics of organizational knowledge. In this context, we define the term organizational knowledge pattern as follows:

An organizational knowledge pattern is a formalization of knowledge for a recurring organizational task abstracting from organization-specific aspects, which is of value for an organizational actor and an asset for an organization.

In the context of this definition, the following characteristics of organizational knowledge patterns (OKP) have to be emphasized:

- OKP are recurring organizational tasks and at the same time abstracting from a specific organization, i.e. like most other kinds of patterns in computer science is the description of the core elements independent from the actual solution for an organization.
- OKP represent organizational knowledge, not individual knowledge, i.e. support the organizational knowledge management, the organizational context for use of knowledge by individuals as opposed to supporting knowledge creation of an individual.
- OKP are an asset of the organization, i.e. are not only a resource as such but capture knowledge about the resource's use. This means they do not only capture the proven practice in a pattern (as for many computer science patterns), but how to use this pattern as resource.
- OKP are expressed in a formalized way, which requires a formal language or at least a structured representation. Thus, OKP are explicit knowledge.
- OKP are of value for an organizational actor.
- The cognitive and technical quality of an OKP is adequate for the stakeholders, i.e. the OKP are developed and described for a defined context of usage with identified stakeholders.

We propose to use the term organizational knowledge pattern in order to emphasize that explicit organizational knowledge is represented and on the other side the technical quality of knowledge patterns is reached.

3 Development and Validation of OKP

This section introduces our proposal for organizational knowledge pattern (OKP) development and validation. We will first present the overall process (section 3.1) and then refine the validation activities (section 3.2).

3.1 Development of Organisational Knowledge Patterns

OKP, as defined in section 2, is a category of artefacts intended for use in enterprise knowledge modeling, which consists of different OKP types, like workflow patterns, task patterns or information demand patterns. Thus, development processes for OKP include the development of the type of OKP – in case this does not yet exist – and development of actual patterns of this type. Since the main intention of OKP is to make organizational knowledge reusable, both the development of pattern types and the development of patterns should involve stakeholders from organizations potentially using the OKP, in order to reach a high "fitness for purpose" of the pattern type and an acceptance of the actual patterns. In this context, one of the guiding principles for our approach of development and validation is stakeholder participation and validation of results from a pragmatic perspective.

The proposed approach for OKP development resembles Boehm's famous "spiral model" [8] for development of software systems and was inspired by this work. Advantages attributed to the spiral model are early validation and continuous improvement of artefacts developed in the process. From our perspective, these characteristics are very useful for collecting feedback from stakeholders in the enterprises under consideration and for ensuring a high accuracy of the knowledge captured in OKP.

Our proposal is to develop organizational knowledge patterns in an iterative way consisting of alternating development and validation steps for both, pattern type and actual patterns. Figure 1 illustrates the overall approach.

In this context, the initial development of a pattern type (step 1; marked as "new type" in figure 1) should primarily focus on the characteristics of the new pattern type and the structure of the pattern representation. Such initial development often is anchored in experiences with certain kinds of organizational knowledge and tries to separate the reusable structures of a solution from the specific solution parts. Next step should be the validation of the initial structure (step 2; "initial struct." in figure 1) consisting of checking internal consistency and soundness. The next step recommended is to apply the new pattern type for capturing an initial and preferably "obvious" and simple pattern example, i.e. the next step is developing an actual pattern (step 3, "simple pattern"). Afterwards, the initial pattern also has to be

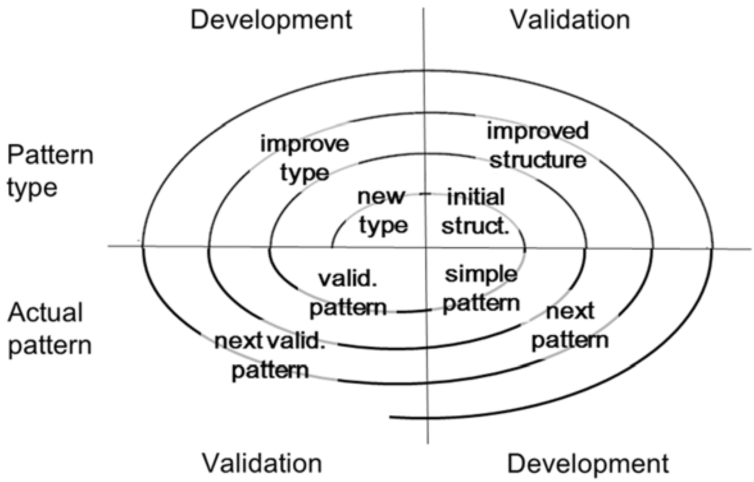


Fig. 1. Iterative process of development and validation of organizational knowledge patterns

validated by checking practical applicability and internal consistency (step 4, "valid. pattern"). Steps 1 to 4 form the first iteration, which was often is performed in a lab environment without involvement from actors outside the research team.

At the beginning of the second iteration focus has to be on improving the definition of the new OKP type, i.e. another development step is necessary (step 5, "improve type"), which has to be based on experiences from developing and validating the first actual pattern (steps 3 and 4). The improvement of the pattern type again has to be followed by the validation for the improved version (step 6, "improved structure"). Steps 7 ("next pattern") and 8 ("next valid. pattern") concern development and validation of the next pattern. A more detailed discussion of the validation activities follows in section 3.2.

3.2 Validation of Organisational Knowledge Patterns

The validation of OKP type and actual patterns has to be performed including stakeholders from future user organizations, and encompassing both theory and practice. Among the many scientific approaches for validating qualitative research results, we base our proposal for validation activities to be performed for OKP on the work of Lincoln and Guba [9, p. 289 ff.] on "naturalistic inquiry". On the one hand, we distinguish between theoretical and practical validation. Theoretical validation means assessing an approach within the theories of the domain the approach is part of or supposed to contribute to. In the context of OKP validation, this means to assess the soundness, feasibility, consistency within the body of knowledge in, for instance, computer science and information systems. Practical validation encompasses all kinds of application of the pattern for validation purposes, which requires defined procedures and documenting results. This could be simple lab examples illustrating the approach, controlled experiments in a lab setting, application in industrial cases, etc.

On the other hand, we consider the context of validation and distinguish between validation by the developers of the approach in their internal environment, validation by the developers outside the internal environment, and validation by other actors than the developers. Combining these two perspectives leads to a two by three matrix, which is depicted in table 1. The cells of this table show typical ways of validation for the different combinations of the two perspectives.

Table 1. Validation steps for OKP

	Theory	**Practice**
Internal, development team	Validation against state of research, internal consistency checks	Prototype implementation for checking feasibility, test in lab environment
External, in validation context	Peer-review of publications describing approach and concepts, comparison to known best practices of the domain.	Case studies with application partners using the artifacts for evaluation purposes Application of the developed artifacts in cooperation / under instruction from developers
External, in application context	Development of extensions or enhancements of the concepts and approaches by external actors Application of the artifacts for creation of new theoretical knowledge Comparison with related approaches	Use of the artifacts developed (e.g. algorithms, methods, software components) for solutions

Using the above matrix, the different iterations of the pattern development described in section 3.1 should proceed from theory to practice and internal to external validation. Thoroughly validated OKP types will include all parts of the matrix and involve several iterations.

The key characteristics of the proposed approach for OKP development and validation can be summarized as follows:

- Tight integration of OKP type development and validation of actual OKP: Since OKP have to be an asset for the organization and of value for at least one stakeholder, the results of validating a pattern can lead to significant changes in the OKP type. If – to take one example – patterns are considered not applicable due to some missing information, the kind of this information might have to be added already on type level.
- Combination of validation from theoretical and practical viewpoints: Validation from theoretical viewpoint is important for achieving high quality of OKP as artefact in solution development processes, i.e. to facilitate transformation or further development to IT-solutions. Validation in practice is important for ensuring pertinence for problems at hand.

- Combination of internal and external validation: OKP are supposed to be used for real problems in real enterprises, i.e. internal validation only by the development team or by researchers is not sufficient and has to be complemented by validation efforts from "external" actors or practitioners.
- Iterative development in alternating elicitation and validation steps: To derive a pattern from existing experiences or enterprise knowledge models in one cycle usually will not lead to an adequate level of maturity. Patterns capture the core structures of solutions, which usually emerge after several validation and improvement steps. High maturity contributes to acceptance for practical use.

4 Case Study: Information Demand Patterns

In order to illustrate our approach proposed in section 3, we will discuss development and validation of information demand patterns as an example. The section will briefly introduce the concept of information demand patterns (4.1) and discuss the development and validation process of these patterns (4.2).

4.1 The Concept of Information Demand Patterns

The concept of information demand pattern originates from work in the research and development project Information Logistics for SME (infoFLOW). infoFLOW included seven partners from automotive supplier industries, IT industry and academia. The objectives were to develop a method for information demand analysis [10] and to identify recurring elements in information demand, i.e. patterns of information demand.

The general idea of information demand patterns (IDP) is similar to most pattern developments in computer science: to capture knowledge about proven solutions in order to facilitate reuse of this knowledge. The term information demand pattern is defined as follows:

An information demand pattern addresses a recurring information flow problem that arises for specific roles and work situations in an enterprise, and presents a conceptual solution to it. [12]

An information demand pattern consists of a number of essential parts used for describing the pattern: pattern name, organisational context, problems addressed, conceptual solution (consisting of information demand, quality criteria and timeline), and effects. These parts will be described in the following.

The *pattern name* usually is the name of the role the pattern addresses.

The *organisational context* explains where the pattern is useful. This context description identifies the application domain or the specific departments or functions in an organisation forming the context for pattern definition.

The *problems* of a role that the pattern addresses are identified. The tasks and responsibilities a certain role has are described in order to identify and discuss the challenges and problems, which this role usually faces in the defined organisational context.

The *conceptual solution* describes how to solve the addressed problem. This includes the i*nformation demand* of the role, which is related to the tasks and responsibilities, a t*imeline* indicating the points in time when the information should be available, and q*uality criteria* for the different elements of the information demand. These criteria include the general importance of the information, the importance of receiving the information completely and with high accuracy, and the importance of timely or real-time information supply.

The *effects* that play in using the proposed solution are described. If the different elements of the information demand should arrive too late or are not available at all this might affect the possibility of the role to complete its task and responsibilities. Information demand patterns include a description of several kinds of effects:

- potential economic consequences;
- time/efficiency effects (i.e. whether the role will need more time for completing the task or will be less efficient);
- effects on increasing or reducing the quality of the work results;
- effects on the motivation of the role responsible;
- learning and experience effects;
- effects from a customer perspective.

The above parts of a pattern are described in much detail in the *textual description* of the pattern. Additionally, a pattern can also be represented as a *visual model*, e.g. a kind of enterprise model. This model representation is supposed to support communication with potential users of the pattern and solution development based on the pattern. An example for an actual pattern for the role of “Material Specification Responsible” in a manufacturing enterprise can be found in [12].

4.2 Development and Validation of Information Demand Patterns

The development and validation of the concept of information demand pattern in general and of actual information demand patterns for specific organizational roles followed the approach proposed in section 3, i.e. it was performed in an iterative way consisting of various cycles, including both information demand pattern type and the actual patterns.

The concept of information demand patterns is inspired by other work in computer science. Thus, the initial work on development of the concept “information demand pattern” focused primarily on the structure of such patterns based on the literature in the field. The validation of the initial structure consisted of checking this structure for internal consistency and soundness, and using it for capturing a very simple pattern example. The elaboration of the first actual pattern using the initial pattern structure consisted of selecting a simple but sufficiently complex organizational role and developing the content of the pattern for the selected role. Afterwards, the initial content of the pattern again was validated by checking internal consistency and completeness, and by using own experiences. The above steps formed the first iteration, which was performed in a lab environment without involvement from actors outside the research team.

The second iteration started with the next development phase for the pattern concept, which was based on experiences from elaborating and validating the first actual pattern (from iteration 1). Improvement of the pattern structure and the following step of validation for the improved version now also involved additional actors, namely the industrial partners of the infoFLOW project. Later iterations included development and validation activities involving actors outside the project team and outside academia.

The validation steps performed for information demand patterns can be summarized using the matrix introduced in section 3.2. This summary is presented in table 2 and shows the following situation: In the infoFLOW project, the information demand pattern concept and selected information demand patterns were evaluated by 15 informants from academia and research institutes. Furthermore, the approach has been applied and evaluated by externals in at least one practical case, which is documented in a master thesis. Other cases are under development supported by a web-portal offering patterns and a handbook for the method developed.

Table 2. Validation status of information demand patterns

	Theory	**Practice**
Internal, development team	Structure of information demand patterns was validated for soundness and consistency against state-of-the-art in computer science; the patterns were evaluated by members of the development team for understandability and completeness	The structure of information demand patterns was used for developing actual patterns; selected patterns were applied for improving a research organization.
External, in validation context	Both, structure and actual demand patterns, were presented at research conferences for validation purposes (e.g. [11]).	Both, structure and actual demand patterns, were presented at industrial events for validation purposes
External, in application context	The structure of information demand patterns was taught in different courses at universities [18]. The course participants used the structure for own pattern development.	The structure of information demand patterns was documented and transferred to selected companies. The pattern development at these companies is ongoing.

5 Discussion

The approach of OKP development and validation proposed in section 3 was applied for information demand patterns (see section 4) and proved both feasible and useful. But is this approach also applicable for other kinds of OKP? This section will discuss this question by investigating two other patterns types and their development and validation processes: task patterns and ontology design patterns. These two types were selected since information about their development and validation process is publicly available. Furthermore, task patterns show all characteristics of OKP defined in section 2.2, but ontology design patterns do not[1]. This might expose differences when discussing the OKP development and validation approach for these two pattern types.

Task patterns are a result of the EU-FP6 project MAPPER [13]. In this project, collaborative engineering was supported by adaptable models capturing best practices for reoccurring tasks in networked enterprises. The term "task patterns" was introduced for these adaptable visual models, as they are not only applicable in a specific company, but are also considered relevant for other enterprises in the application domain under consideration. Task pattern in this context is defined as "self-contained model template with well-defined connectors to application environments capturing knowledge about best practices for a clearly defined task" [14]. In this context, self-contained means that a task pattern includes all POPS*[2] perspectives, model elements and relationships between the model elements required for capturing the knowledge reflecting a best practice. Model template indicates the use of a well-defined modeling language and that no instances are contained in the task patterns. Connectors are model elements representing the adaptation of the task pattern to target application environments.

The term *ontology design pattern* was introduced by Gangemi [15] and Blomqvist & Sandkuhl [16]. An ontology pattern is a set of ontological elements, structures or construction principles that intend to solve a specific engineering problem and that recur, either exactly replicated or in an adapted form, within some set of ontologies or is envisioned to recur within some future set of ontologies. Two types of patterns are of particular interest: *structural-* and *content* patterns. *Structural patterns* deal only with the logical structure of the ontological elements but not with the actual ontology represented by these. A structural pattern is only a logical vocabulary with an empty signature, so no actual concepts and relations or other axioms are actually present. *Content patterns* are a specialization of structural patterns, since they both constrain the logical structure of how the solution to the problem should be modeled and set requirements on the ontological content. Content patterns are instantiations, and possibly combinations, of structural patterns where the signature is no longer empty.

The core characteristics of the OKP development and validation approach presented in section 3.1 will be used to investigate the appropriateness of these characteristics for the task pattern and ontology design pattern development. In this context, the core question is not, whether task patterns and ontology design patterns can be developed with this approach, but whether these criteria are *essential* for high quality and maturity of pattern types and results. For the sake of brevity, this discussion will be summarized in the following table.

---

[1] Ontology design patterns capture proven practice of ontology design, but they do not capture the context of the ontology pattern use. Furthermore, ontology design patterns are of value for individual ontology engineers, but not necessarily an asset for the organization as such, since they are very specific and may be used for many ontologies, but just once in an organization.

[2] The POPS* perspectives include the enterprise's processes (P), the organization structure (O), the products developed (P), the IT-systems used (S) and other aspects deemed relevant when modeling (*) [17].

Table 3. Validity of OKP characteristics for task patterns and ontology design patterns

Pattern type *Characteristic*	**Task Pattern**	**Ontology Design Pattern**
Integration of type development and pattern development	TP concept emerged while identifying best practices and was changed during first pattern developments. Later, the pattern structure served as blueprint for many more patterns. Integration of both steps was essential.	The pattern types and pattern structures were proposed based on illustrative examples. Refinements in the type definition were often due to discussion in an academic context, but sometimes also made based on experiences.
Validation in theory and practice	Theoretical validation was important for ensuring meta-model compatibility. Validation in practice was essential for contributing to problem solving.	Theoretical validation is essential for compatibility to underlying logical design pattern. Validation in practice was essential to reach acceptance by ontology engineers.
Internal and external validation	Development team included stakeholders from research and industry, and resulted in high maturity. Application in other industries improved transferability. Thus, internal and external validation was essential.	Development of patterns primarily happened within academic context. External validation is still ongoing.
Iterative development and validation	Development of actual patterns and development of pattern structures were tightly related and not possible in just one run.	Only in some cases of importance: experienced ontology engineers can propose ontology design patterns based on previous experiences, which basically are confirmed without changes during validation.

The above table shows that all characteristics of the proposed approach have to be considered essential for task patterns, i.e. the approach is not only valid for information demand patterns (as shown in section 4), but also for task patterns. However, when it comes to ontology design patterns, iterative development and validation is not as important as for task patterns, and internal and external validation is recommendable, but not essential. This indicates that the proposed approach has limitations for pattern types which are primarily used in engineering or solution development contexts.

6 Summary and Future Work

This paper investigated the term of organizational knowledge patterns and their development and validation process. The core characteristics of OKP include that they capture organizational knowledge including the context of its use and that OKP only can be considered as mature if they are of value and an asset for the organization and not only for individuals.

In order to achieve these characteristics, we propose a development and validation approach for OKP tailored for use in applied research, i.e. in a mixed academic and enterprise context. The key features of this approach are an iterative development process with alternating elaboration and validation phases. The validation phases include a combination of internal and external validation in both theory and practice.

This approach proved suitable for information demand patterns and task patterns. For ontology design patterns, limitations of the approach were detected in that the approach still was feasible, but some of the activities considered essential were not required.

Iterative development of artefacts in computer science is not new; neither is a combination of validation in theory and practice. However, the combination of both approaches suits the needs of OKP very well, since their applicability in industrial contexts and acceptance by business stakeholders is essential.

Future work will have to include further elaboration of the approach towards a method, i.e. a more detailed definition of how to set up and implement the different steps of iterative development, how to capture the results of development and validation, and what aspects or concepts to focus on in each and every step. Furthermore, the concept of organizational knowledge patterns has to be developed further, e.g. by providing more examples for OKP types, evaluating them in detail in order to learn about improvement potential and limitations of this approach.

Acknowledgements. Part of this work was funded by the R&D project “Demand Patterns for Efficient Information Logistics (infoFLOW-2)” financed by the Swedish Knowledge Foundation with grant 2009/0257. The author wishes to thank all infoFLOW co-workers involved in pattern development and validation.

References

1. Vernadat, F.B.: Enterprise Modelling and Integration. Chapman & Hall, Boca Raton (1996)
2. Alexander, C., et al.: A Pattern Language, Oxford (1977)
3. Clark, P., Thompson, J., Porter, B.: Knowledge Patterns. In: Cohn, A.G., Giunchiglia, F., Selman, B. (eds.) KR 2000: Principles of Knowledge Representation and Reasoning. Morgan Kaufman, San Francisco (2000)
4. Staab, S., Erdmann, M., Maedche, A.: Engineering Ontologies using Semantic Patterns. In: O’Leary, D., Preece, A. (eds.) Proceedings of the IJCAI 2001 Workshop on E-business & The Intelligent Web, Seattle (2001)
5. Puppe, F.: Knowledge Formalization Patterns. In: Proceedings of PKAW 2000, Sydney, Australia (2000)
6. Van der Aalst, W.M.P., ter Hofstede, A.H.M., Kiepuszewski, B., Barros, A.P.: Workflow Patterns. Distributed and Parallel Databases 14, 5–51 (2003)
7. Schümmer, T., Lukosch, S.: Patterns for Computer-Mediated Interaction. Wiley & Sons, Chichester (2007), ISBN: 978-0-470-02561-1
8. Boehm, B.W.: A spiral model of software development and enhancement. IEEE Computer 21(5), 61–72 (1988)
9. Lincoln, Y., Guba, E.: Naturalistic Inquiry. Sage Publications, Thousand Oaks (1985)

10. Lundqvist, M., Sandkuhl, K., Seigerroth, U.: Modelling Information Demand in an Enterprise Context: Method, Notation and Lessons Learned. International Journal Systems Modeling and Design 3(1) (2011)
11. Sandkuhl, K.: Improving Engineering Change Management with Information Demand Patterns. In: 8th International Conference on Product Lifecycle Management, Eindhoven, The Netherlands. Inderscience Enterprises, July 12-14 (2011)
12. Sandkuhl, K.: Supporting Collaborative Engineering with Information Supply Patterns. In: Euromicro PDP 2010. IEEE/CS, Pisa (2010)
13. Johnsen, S., Schümmer, T., Haake, J., Pawlak, A., Jørgensen, H., Sandkuhl, K., Stirna, J., Tellioglu, H., Jaccuci, G.: Model-based Adaptive Product and Process Engineering. In: Rabe, M., Mihók, P. (eds.) New Technologies for the Intelligent Design and Operation of Manufacturing Networks. Fraunhofer IRB Verlag, Stuttgart (2007)
14. Sandkuhl, K.: Capturing Product Development Knowledge with Task Patterns: Evaluation of Economic Effects. Quarterly Journal of Control & Cybernetics (1) (2010)
15. Gangemi, A.: Ontology design patterns for semantic web content. In: Gil, Y., Motta, E., Benjamins, V.R., Musen, M.A. (eds.) ISWC 2005. LNCS, vol. 3729, pp. 262–276. Springer, Heidelberg (2005)
16. Blomqvist, E., Sandkuhl, K.: Patterns in Ontology Engineering – Classification of Ontology Patterns. In: Proc. 7th International Conference on Enterprise Information Systems, Miami, USA (May 2005)
17. Lillehagen, F.: The Foundations of AKM Technology. In: Proceedings 10th International Conference on Concurrent Engineering (CE) Conference, Madeira, Portugal (2003)
18. Sandkuhl, K.: Validation and Use of Information Demand Patterns in Higher Education. In: BIS 2010 International Workshops. LNBIP, vol. 57. Springer, Heidelberg (2010)

Analysis of Dynamic Interactions with External Parties During Maintenance of ERP Systems

Igors Mikulovs[1] and Jānis Grabis[2]

[1] Tele2 AB
Tele2 Shared Service Center
41B Mukusalas Street, Riga, Latvia
igors.mikulovs@tele2.com
[2] Institute of Information Technology
Riga Technical University
Kalku 1, Rīga, LV-1658, Latvia
grabis@iti.rtu.lv

Abstract. The paper investigates maintenance practices of Enterprise Resource Planning (ERP) systems with emphasis on interactions between on-site maintenance team and third parties. A case study approach is used to identify maintenance activities in ERP systems, to classify them into maintenance categories, and to characterize dynamics and frequency of interactions with the third-party. Multiple deployments of the same ERP system are used in the investigation. Each deployment is in a different stage of its life-cycle what allows analyzing maintenance activities with regards to system's maturity. Five common maintenance activities (corrective, adaptive, perfective, preventive, user support) are considered while main attention is devoted to a recently introduced category pertaining to 'communication, coordination and knowledge exchange with external parties' and named 'external parties'. Interviews with maintenance staff are used to identify maintenance tasks and maintenance requests database is used for numerical analysis of maintenance requests. The results show that majority of maintenance tasks are performed by the on-site team and maintenance requests escalated to the third party require longer implementation lead times.

Keywords: Enterprise resource planning, external vendor, software maintenance.

1 Introduction

Enterprise resource planning (ERP) applications were one of the fastest growing and most profitable areas of the software industry during the late 1990s [1]. Companies worldwide have invested billions of dollars in implementing these systems. The adoption of ERP systems has been the focus of substantial research in recent years [2]. Although early research has covered many aspects of ERP implementation, it has focused mainly on the earlier stages of the ERP lifecycle [3]. As with other types of information systems, many maintenance activities must be carried out, and issues need to be resolved after an ERP system has become operational [4]. Examples of such activities include enhancements, user support, and system upgrades.

J. Grabis and M. Kirikova (Eds.): BIR 2011, LNBIP 90, pp. 217–230, 2011.

This research focuses on the post-implementation or maintenance stage of ERP systems. Prior empirical research has shown that many important issues and problems arise during this stage [5]. However, previous studies look at the ERP system in isolation, role of external vendor and interactions with it have been sparsely researched. The aim of this study is to emphasize the role of external party and to expand previous studies by looking at additional factors related to the vendor during ERP system maintenance. Thus, it addresses maintenance activities that take place after an ERP system goes live, classifies these activities into meaningful maintenance categories, defines type of interactions taking place between the ERP user and the vendor, and analyzes frequency of these interactions. Particular research questions are:

1. What are the maintenance activities pertaining to ERP?
2. How maintenance activities can be categorized?
3. What is the role of external party in operational ERP system?
4. How external party activities interact with regular ERP system maintenance activities? What are dependencies?

Definition of interactions between on-site maintenance team of the ERP system and the vendor of the ERP system is the main contribution of this research. This study is relevant to both research and practitioner communities. It contributes to knowledge on the post-implementation stage of ERP systems — an area where little research has been conducted. The findings increase our understanding of how maintenance of software packages, particularly large software packages, differs from that of custom-built information systems. Such understanding is important and necessary for effective information systems management of large software packages.

This paper provides a brief review of software maintenance literature, describes the research methodology employed in this investigation, presents analysis of the results and findings, and provides discussion and conclusions.

2 Background and Related Work

ERP systems are complex systems, and their life-cycle can be divided into two major phases: implementation and maintenance. The important feature of these systems is that maintenance efforts are comparable with implementation efforts.

2.1 Enterprise Resource Planning Systems

ERP systems integrate internal and external management information across an entire organization, embracing finance/accounting, manufacturing, sales and service, CRM, etc. ERP systems automate this activity with an integrated software application. Its purpose is to facilitate the flow of information between all business functions inside the boundaries of the organization and manage the connections to outside stakeholders [6].

Worldwide ERP software sales grew by 14 percent in 2006 to $28.8 billion, according to a recent report from IT analyst firm AMR Research. Market for ERP has been steadily escalating with sales of $40 billion in 2009 and $43 billion in 2010; the latest report, in fact, predicts the global ERP market will reach $45 billion in 2011, and $50 billion by 2015. Not only large companies, but also mid-size companies, are adopting ERP [7]. This has made ERP software the fastest growing segment in the software marketplace. Currently the dominant vendors in the ERP software market are SAP, Oracle and Microsoft [8].

Being packaged software, ERP is designed with a 'class' of organizations in mind and is claimed to incorporate 'best business practices.' ERP systems require standardization of data and business processes across an organization to enable integration [8]. Although E7RP systems are customizable, they are difficult and costly to adapt to unique organizational procedures. Often an organization's business processes must be modified to fit the system. Reengineering existing business processes is a critical implementation concern and a key success factor of ERP implementation [9].

The adoption of commercial software packages differs in many ways from the adoption of in-house developed software: new IT skills are needed in the organization, there is a greater dependency on external vendors, ERP software needs to be integrated with a combination of hardware, operating systems, database management systems software and telecommunications suited to the characteristics of the specific organization, and, in most cases, the ERP software will need to interface with legacy systems [10].

The implementation concerns of ERP do not end once the system becomes operational. As indicated by Davenport [11], 'an enterprise system is not a project; it's a way of life'. He argues against the traditional assumption of treating ERP as a project that has a termination date. Users need on-going support and organizations face a variety of issues such as fixing problems, upgrading to new versions of the software, and managing organizational performance with the system to achieve desired benefits. Since ERP is continually evolving and is a fairly new phenomenon, little research has been done to study maintenance of these systems.

2.2 Maintenance

Software maintenance is defined as the modification of a software system after delivery to correct faults, improve performance, or adapt to a changed environment [12]. The literature on software maintenance covers various categorizations of maintenance activities. One of the classifications that is the most frequently cited in the maintenance literature was proposed by Lientz and Swanson [13], which classifies maintenance activities into three main categories.

1. Corrective maintenance: work to correct errors in design, coding, and implementation.
2. Adaptive maintenance: work carried out to satisfy changes in the processing or data environment and to meet new user requirements.
3. Perfective maintenance: work to enhance processing efficiency, performance, or maintainability and to better meet user requirements.

Various authors have introduced additional categories to this classification or have expanded the number of activities that fall under the above-mentioned categories. Burch and Grupe [14] introduced another category termed 'preventive maintenance' which refers to periodic inspection of systems to anticipate problems. Nah [15] stated

‘The maintenance personnel may detect defects that may not require immediate attention, but, if left uncorrected, could significantly affect either the functioning of the system or the maintainability of the system in the future.

User support is another category proposed by Abran and Nguyenkim [16]. Their research shows that user support takes up 24% of the total time spent on systems maintenance. Hirt and Swanson [17] suggest that users play a more important role in maintenance of ERP systems than traditional systems. While user training is also carried out at the early stages of the implementation process, there is always a need for continuous training as new users and new functionality are added to the system.

The findings from the literature are inconsistent with regard to the amount of effort spent on activities in each maintenance category. According to Barry et al. [18], enhancement effort is the largest category in the software maintenance lifecycle. Results from a study by Abran and Nguyenkim [16] suggest that corrective maintenance is the category where the greatest amount of effort is invested (35%). Zvegintzov [19] compared the findings of three different studies concerning the amount of effort spent on each maintenance category. His results suggest that adaptive maintenance is the largest category. Since maintenance has long been recognized as the most costly activity in the software lifecycle, the frequency, distribution, and trend of activities carried out during software maintenance warrant further research and improved understanding.

Although the literature recognizes that information systems staff, users, and systems are three main parties concerned with the maintenance of custom-built information systems, with ERP, ‘external parties’ become important as well [15]. In contrast to in-house developed information systems, ERP systems are developed by outside vendors and other third parties are also involved in the implementation process. These third parties are likely to be responsible for some part of the maintenance burden. External party as a software maintenance activity specific to ERP systems was also described by Nah [15]. Despite of that, interactions with third-parties during the maintenance process have not been sufficiently explored in research literature.

3 Research Methodology

A single case study approach [20] was undertaken in this paper. However, the specific feature of this case study is that maintenance practices are studied for several implementations of the same ERP system at sub-divisions of the company under consideration, and each implementation is at a different system’s life-cycle stage. That allows exploring maintenance practices throughout the life-cycle. The primary objective of the case study is to uncover maintenance practices used with emphasis on dynamic interactions with a third-party, which is responsible for some of maintenance tasks.

An object of this case study is maintenance of the ERP system at a major European multinational telecommunication company Tele2 AB. This company has about 30 million customers in 11 countries. It serves as a fixed-line telephone operator, cable television provider, mobile phone operator and Internet service provider. Tele2 has implemented several ERP and Billing software packages across the business. Systems observed in this study are implemented in Lithuania, Latvia, Estonia and Croatia and are based on the same ERP software package (i.e., there are four implementation variants of the same system). They serve almost 2 million customers in four countries, providing CRM, Billing and Financial platforms. The base ERP system is developed by a third party. Furthermore, this vendor has about 40 similar software package installations across their customers. The software package has 15 years old history and currently is being actively developed and supported. Majority of maintenance activities are performed by the company itself though the vendor of the ERP system is involved if necessary [21].

The observed installations currently are in the following software lifecycle stages:

- 1st system instance – growth stage, vendors branch is kept;
- 2nd system instance – decline stage;
- 3rd system instance – maturity stage;
- 4th system instance– maturity stage, many in-house developments.

Since this study looks specifically at ERP maintenance activities, we contacted and interviewed a support and implementation manager, who had the main responsibility for overseeing the ERP systems. We interviewed a change/release manager, a problem manager and system managers for all four ERP system instances. These three individuals are involved in interaction with the third party, coordination of new releases, helpdesk communication activities, and new patch deliveries. The author was participating in implementation of the ERP systems in one of the countries and has experience from pre-study, ongoing implementation, user support, to maintenance of the system.

To learn about perfective maintenance activity within the organization, the database administrator was interviewed. This individual is involved in possible perfective tasks, database improvements, hardware improvements across all four system instances.

This case study consisted of three main phases — preliminary interviews, a open-ended questionnaire with follow-up interviews, and exploration of maintenance tasks statistics. In the first phase, interviews were carried out to gain insights into the various ERP implementation and maintenance processes. In the second phase, an open-ended questionnaire was distributed to the interviewees as part of the case study protocol. The questionnaire comprised the following two questions:

1. Please describe maintenance tasks and activities that have taken since the ERP system went live.
2. What is approximate amount of effort spent on tasks in each category, time for resolution and number of cases of external party involvement for each category?

The objective of the second phase was to identify a list of tasks undertaken during the maintenance stage and to determine frequency of external party involvement. The second phase also included registering and categorizing different maintenance tasks performed by the company using the ERP system. This categorization is based on the traditional categorization discussed above [12] with the addition of the external party category [15].

The maintenance tasks identified in the second phase were consolidated across the all four instances and used to develop the structured questionnaire, which was administered to the interviewees in the third phase. In the third phase, statistics of registered maintenance tasks in year 2010 for all instances were analyzed.

4 Analysis and Findings

The open preliminary interviews provided basic understanding of overall ERP system implementation tasks in the organization and insights into the maintenance activities of ERP. The main problems faced during the interviews were naming and classification of maintenance activities as well as grouping them into the final list by categories. The follow-up interviews were useful to clarify and resolve naming misunderstandings.

In the second phase, 21 maintenance tasks were identified within the organization, they were grouped in defined categories (corrective, adaptive, perfective, preventive, user support, external vendor) based on description of the tasks, comments given by the respondents and definition of existing categories suggested in literature. Table 1 shows the list of maintenance categories and tasks.

The maintenance of ERP software was found to comprise some tasks not typically performed during the maintenance of custom-build software, most of the tasks fit traditional categories of maintenance: corrective, adaptive, perfective, preventive and user support.

External parties are identified as a significant category of maintenance activities for the ERP systems, comparing with in-house developed systems. This category includes: vendor communicating, consultancy, new releases and version applying, vendor's help desk and permanent issue resolutions, change request analyses. Coordination and interaction with external parties play indirect or supporting role across the rest 5 categories. Hirt and Swanson [17] have also proposed a similar idea of including the consideration of 'external parties' in ERP maintenance. They acknowledged the need for changes/additions in the ERP context and indicated that 'it is perhaps not surprising that the traditional perspective of maintenance needs to be transformed in order to accommodate ERP'.

Nah [15] supported that by quoting: 'Since coordination with external parties and creation of OSS (Operational Support System) notes are largely external in nature, they were grouped under a new category named commutation, coordination, and knowledge exchange with external parties' and abbreviated as 'external parties'.

During the second phase of the case study, typical scenarios of interacting with the external party are also identified. The maintenance interactions are represented as timelines, where one timeline is used to show activities by the company and another one is used to represent involvement of the third-party. These interactions are shown for each of the maintenance categories. Individual maintenance tasks usually serve as interaction triggers. The following diagrams are applicable for all four systems observed.

Table 1. Classification of maintenance tasks into maintenance categories

Maintenance categories	Maintenance Tasks	Description
(1) Corrective maintenance	Internal permanent fix	System configuration, process correction, that can be resolved by IT department
	External permanent fix	Incorporate system patches set by vendor(in case when issue can't be resolved by the IT department)
	Internal temporary fix	Temporary fix performed within IT department as work around, while permanent solution is prepared by software vendor.
(2) Adaptive maintenance	Change implementation	New product, processes and functionality implementation.
	Testing	System integration, regression and system tests after new patches application and configuration changes.
	Release installation	Moving new patches from test environment to production. Coordination of downtimes.
(3) Perfective maintenance	Hardware upgrade	Hardware upgrade in order to improve performance or increase storage space
	DB performance improvements	DB analyses and configuration improvements.
	Version upgrade	System version upgrade.
(4) Preventive maintenance	Monitoring workflows	Monitoring of error logs, average system response times.
	Monitoring DB load	Monitoring DB SQL queries.
	Monitoring storage space	Monitoring table spaces, files spaces and file sizes.
(5) User support	Support Requests	
	Incidents	
	Technical queries	
(6) External parties	Help Desk communication	Coordination work and relation among internal IT department members, vendors, consultations and external user organizations
	Problems reports	Tracking vendor's progress towards resolution of problem reported Bug fixes performed by
	Release deliveries	Tracking vendor's delivered releases, release strategy and delivery plans.
	Change request coordination	Tracking of Change requests towards vendor's software, costs and estimates agreement, purchase order composition
	Technical query	Tracking of user support escalated question and technical enquiries.

In the case of corrective maintenance, three situations are distinguished: 1) a permanent corrective action that could be performed without external party; 2) a permanent corrective action that could be performed only with external party involvement; and 3) a temporary corrective action that could be performed as work around awaiting the external party permanent fix (Figure 1). Third situation was identified during the case study as very frequent because external party fixes often are not treated on the emergency level. According to the interviews, enquiries registered with the external party are resolved 5-7 times longer than internally performed corrective activities [22, 23].

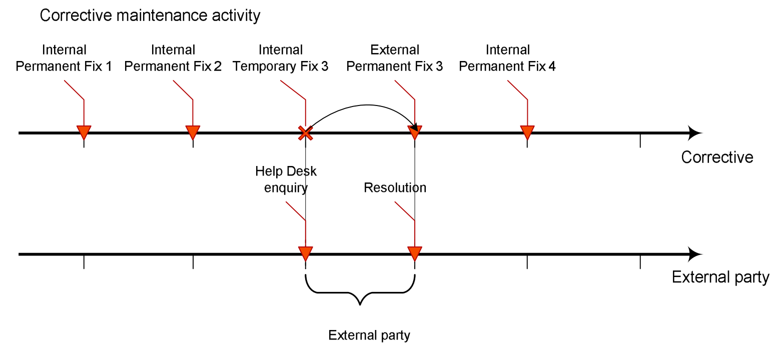


Fig. 1. Interaction within Corrective activities

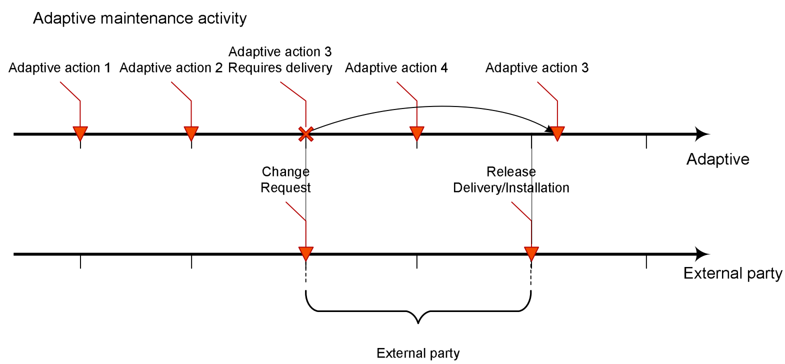


Fig. 2. Interaction within Adaptive activities

In the case of adaptive activities, two situations are distinguished. The first one does not require new functionality to implement changes into the system. The second one requires changes in the code base and is treated as a change request to the ERP software on the vendor's side. According to the interviews, analysis, development and testing of change request registered with the external party registered take 10-15 times longer than internally performed adaptive activity. This can be explained with complexity of changes requested to the vendor, comparing with performed internally.

Resource allocation and SLA (Service level agreement) does not exist for change requests, though, this kind of activities are driven by business case, agreements and required delivery estimates.

In the case of perfective maintenance, two situations are distinguished. The first does not require vendor's interaction and could be performed on site (e.g. hardware upgrade), the second is related to new versions of ERP or applying a patch that increase performance for the overall system or its parts (Figure 3). Patch delivery, testing and installation time estimates are comparably close with internally performed perfective maintenance activity time periods [8].

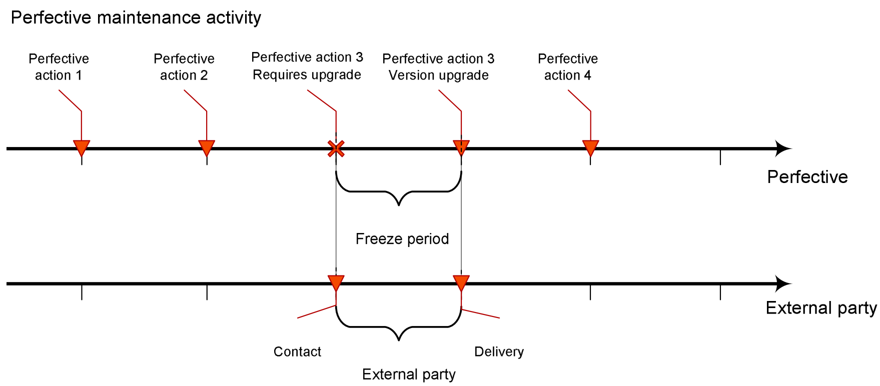


Fig. 3. Interaction within Perfective activities

Preventive activities trigger interactions with the external party activity in the case of vendor's deliveries and a preventive activity is performed as a part of the sanity check (Figure 4). However, regular preventive activities not related to new software releases (daily log monitoring, database load monitoring) are performed on site. Preventive actions time frame triggered by the external party activity (new release or patch) is comparably close to internally performed preventive maintenance activities.

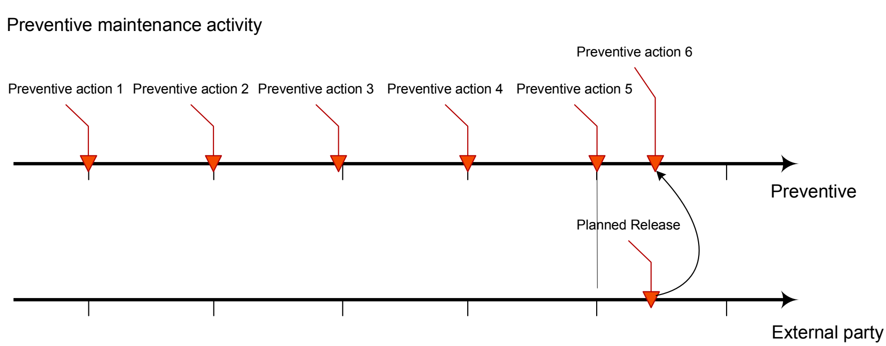


Fig. 4. Interaction within Preventive activities

In the case of user support, two situations are distinguished: 1) ones that require escalation to the vendor's help desk in purpose to consult and fulfill a request with complete information; and 2) ones that do not require escalation and can be supported on site [24] (Figure 5). According to interviews, resolution time for requests escalated to the vendor's help desk is 3-4 times longer than for internal user support activities.

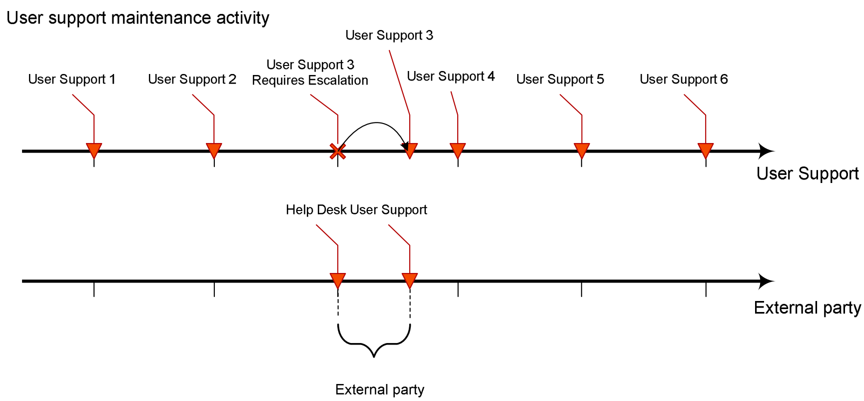


Fig. 5. Interaction within User support activities

An example of interactions shown in Figure 1 is as follows: an error found in the balance calculation module and the calculation formula is coded in the module and cannot be modified. To satisfy customer, manual adjustments in calculations could be performed for particular problematic cases, this temporary action could be performed unless permanent fix for the balance calculation module is not delivered. Adaptive action interactions with the external party shown on Figure 2 are invoked, if a new business requirement cannot be implemented based on existing functionality. For instance, it is necessary to make integration of new sales process in the ERP system, but there is no interface to make such integration. In order to fulfill the business requirement, it is necessary to order new functionality as integration interface from the vendor.

Figures 6 shows the frequency of triggered external party activity comparing with frequency of each category maintenance activities across all four ERP system instances in different lifecycle stages. Figures are built based on statistics and reports provided by the company. Important note is that the 1st system just passed introduction stage. The system experienced upgrade to a new version and it is on rolling from introduction to growth stage. The forth system has many in-house developments, partially it can be named as traditional system, therefore there are less appeals towards external vendor, as significant number of activities and tasks can be resolved internally. Such results emphasize and secrete external vendor activities that inherit particularly for ERP systems and confirm previously performed researches.

The system that is not upgraded due to some certain business limitations, and continues to grow using in-house developments, can be perceived as in an extension stage as a part of the maturity stage. Though, with the scarcity and high turnover rate of ERP skills, in-house customizations can cause threats to the quality of the system, and to the success of ERP at the maturity stage [25].

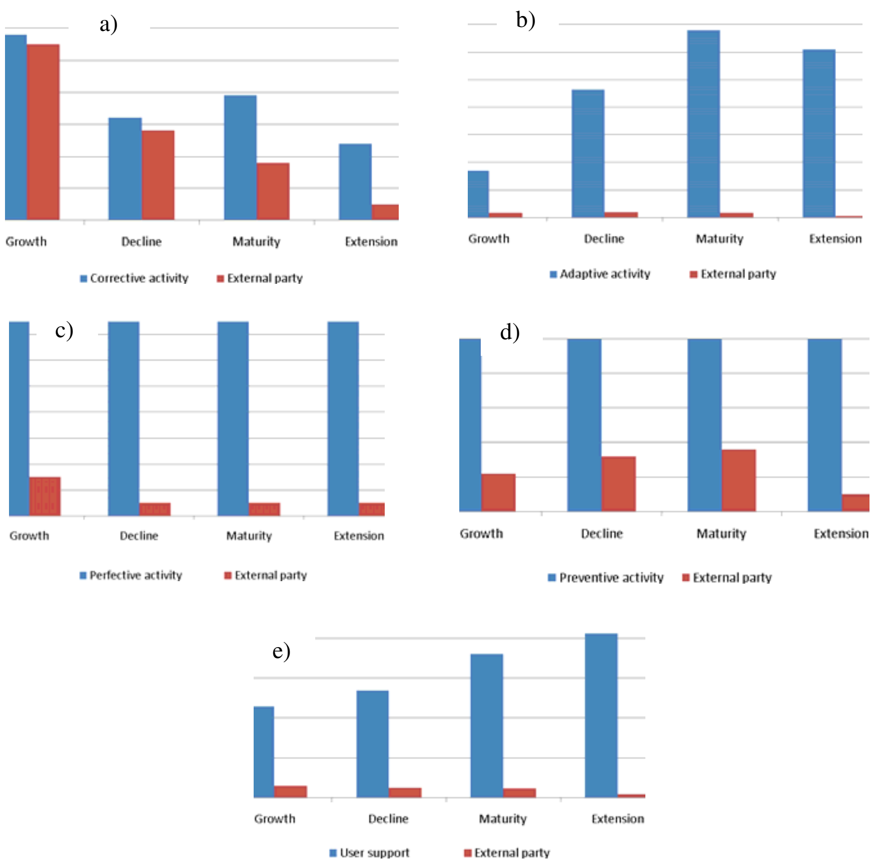


Fig. 6. The relative frequency of maintenance requests escalated to the third party compared to the total number of maintenance requests for a) corrective activities; b) adaptive activties; c) perfective activities; d) preventive activities; and e) user suport activities

The main finding regarding maintenance activities during the life-cycle of the ERP system can be summarized as follows.

Introduction stage. Similar to maintenance of traditional in-house software [16] the main focus during the introduction stage of ERP maintenance is user support. Adaptive, corrective, and preventive maintenance are also important and take up much time and effort.

Growth stage. During the growth stage, all the categories, except perfective maintenance, decline and exhibit similar frequencies. The frequency of these categories continues to decline through the maturity stage.

Maturity stage. Perfective maintenance increases at the maturity stage. Version upgrade is the main activity that falls into this category. According to the representative of Tele2, seven people (about 10% of the ERP maintenance staff) were involved in 4th system version upgrade.

As end-users become familiar with the system, the need for new or better features increases. Upgrade is carried out to satisfy end-user needs, to take advantage of improved functionality, and to overcome pressure from competitive forces to adapt to the latest ‘best business practices’. Otherwise upgrade could be performed whenever new functionality can’t be delivered in patches or releases and common system re-design is necessary in order to satisfy business needs.

Extension stage. This stage cannot be perceived as a self-sufficient stage, and occurs if in-house development is performed instead of upgrading to a new version of the system. That can be perceived as a work around but not as a long-term solution.

5 Discussion and Conclusion

Using the case study from the telecommunication company, the activities and tasks pertaining to ERP maintenance have been identified and classified into six maintenance categories. The classification and frequency of maintenance tasks is generally consistent with previous research on traditional systems maintenance, with some exceptions and novelties.

The study confirms importance of the newly introduced external party category and identifies tasks that are performed within this category. Furthermore, research presents dependencies between regular maintenance and external party field.

Tasks of corrective maintenance trigger involvement of the third party most frequently. Corrective maintenance is the most characteristic to systems in the early introduction stage of the lifecycle. However, in this case many adaptive tasks are performed in mature systems what leads to many following corrective actions. Some of them cannot be resolved internally and should be referred to the vendor.

The study evidences a relative decrease of corrective actions across lifecycle stages, and rapid growth for the system that has recently passed the introduction stage. That confirms findings from previous research that indicates increase of corrective maintenance during the growth and introduction stages.

The study shows that the different categories of maintenance activities have different triggers of external party activity. Adaptive activities can be mostly handled without third party involvement, though corrective activities mostly are related to the vendor. Study provided an additional non self-sufficient software stage as part of the maturity stage - extension, which is explained as system further development using in-house approach avoiding vendors involvement, due to certain business limitations (e.g. costs, time). Significant finding is the fact that problem resolution takes longer time if the external party is involved, that highlights increasing importance of the SLA agreement with the vendor. This could be one of the major reasons for inventing the extension stage.

The study has a number of limitations. One of limitation is that all four observed ERP system instances have the same vendor. Since the vendor plays an important role during the ERP system lifecycle, this may affect the generalizability of the study because different vendors may have different SLA contracts and approaches working with their customers, as well can offer different level of support on the post-implementation stage. This study also does not address cost and resource requirements [25].

In spite of the above limitations, study provides important insights into maintenance process of ERP systems and their interaction with vendors. Future research might consider validation of this study research in other organizations as well as for other types of enterprise applications. It is also important to research more cases of dependencies and triggers for external party activities. It is important to define characteristics of change requests what cause necessity to involve the third party. In future research costs and resource parameters should be included to represent more valid information regarding time of external party responses and resolution periods.

The findings of this study may aid in better management, planning and coordination of ERP system maintenance. With a better understanding of maintenance process IS managers can better allocate appropriate recourses to relevant activities and pick out certain responsible for interaction with external vendor. For instance, a corrective action in many cases should receive permanent fix from the external party, rather than keeping the developed work around internally for a long period, that could cause worse maintainability.

References

1. Sprott, D.: Componentizing the enterprise application packages. Communications of the ACM 43(4), 63–69 (2000)
2. Law, C.C.H., Chen, C.C., Wu, B.J.P.: Managing the full ERP life-cycle: Considerations of maintenance and support requirements and IT governance practice as integral elements of the formula for successful ERP adoption. Computers in Industry 61(3), 297–308 (2010)
3. Botta-Genoulaz, V., Millet, V., Grabot, B.: A survey on the recent research on ERP systems. Computers in Industry 56, 510–512 (2005)
4. Daveport, T.: Putting the enterprise into the enterprise systems. Harvard Business Review 76(4), 121–131 (1988)
5. Law, C.C.H., Ngai, E.W.T.: ERP systems adoption: An exploratory study of the organizational factors and impacts of ERP success. Information & Management 44(4), 418–432 (2007)
6. King, W.: Ensuring ERP implementation success. Information Systems Management 22(3), 83–84 (2005)
7. Woodie, A.: AMR Research Bullish on ERP Software Market (July 23, 2007), http://www.itjungle.com/tfh/tfh072307-story08.html
8. Bailor, C.: For CRM, ERP, and SCM, SAP Leads the Way, http://www.destinationcrm.com/articles/default.asp?ArticleID=6162 (retrieved March 29, 2007)
9. Nah, F., Lau, J., Kuang, J.: Critical factors for successful implementation of enterprise systems. Business Process Management Journal 7(3), 285–296 (2001)
10. Markus, M.L., Tanis, C.: The enterprise system experience-from adoption to success. In: Zmud, R.W. (ed.) Framing the Domainsof IT Management: Projecting the Future.. Through the Past, pp. 173–207. Pinnaflex Educational Resources, Cincinnati (2000)
11. Davenport, T.H.: Living with ERP. CIO Magazine (December 1, 1998), http://www.cio.com/archive/120198think.html

12. ANSI/IEEE, Standard 729: An American National Standard/IEEE Standard Glossary of Software Engineering Terminology. Institute of Electrical and Electronic Engineering: New York NY (1983)
13. Lientz, B., Swanson, E.: Problems in application software maintenance. Communications of the ACM 24(11), 763–769 (1981)
14. Burch, J., Grupe, F.: Improved software maintenance management. Information Systems Management 10(1), 24–33 (1993)
15. Nah, F., Faja, S., Cata, T.: Characteristics of ERP software maintenance: a multiple case study. Journal of Software Maintenance and Evolution: Research and Practice 13, 399–414 (2001)
16. Abran, A., Nguyenkim, H.: Analysis of maintenance work categories through measurement. In: Proceedings Conference on Software Maintenance, pp. 104–113. IEEE Computer Society Press, Los Alamitos CA (1991)
17. Hirt, S.G., Swanson, E.B..: Maintaining ERP: Rethinking relational foundations. Working Paper 02-99. The Anderson School at University of California, Los Angeles, Los Angeles CA, (1999)
18. Barry, E.J., Kemerer, C.F., Slaughter, S.A.: Toward a classification scheme for software maintenance activities. In: Proceedings of the Fifth Americas Conference on Information Systems, pp. 726–727. Association for Information Systems, Atlanta (1999)
19. Zvegintzov, N.: Real maintenance statistics. Software Maintenance News 9(2), 6–9 (1991)
20. Benbasat, I., Goldstein, D., Mead, M.: The case research strategies in studies of information systems. MIS Quarterly 11(3), 369–386 (1987)
21. Gable, G., Chan, T., Tan, W.: Large packaged application software maintenance: a research framework. Journal of Software Maintenance and Evolution: Research and Practice 13(6), 351–371 (2001)
22. Gosain, S., Lee, Z., Kim, Y.: The management of cross-functional inter-dependencies in ERP implementations: emergent coordination patterns. European Journal of Information Systems 14(4), 371–387 (2005)
23. Hirt, S., Swanson, E.: Emergent maintenance of ERP: new roles and relationships. Journal of Software Maintenance and Evolution: Research and Practice 13(6), 373–387 (2001)
24. Scott, J.: Post-implementation usability of ERP training manuals: the user's perspective. Information Systems Management 22(2), 67–77 (2005)
25. Beard, J., Sumner, M.: Seeking strategic advantage in the post-net era: viewing ERP systems from the resource-based perspective. Journal of Strategic Information Systems 13, 129–150 (2004)

Discovering of Users' Interests Evolution Patterns for Learning Goals Recommendation

Witold Abramowicz, Jacek Małyszko, and Dawid Grzegorz Węckowski

Poznań University of Economics,
Al. Niepodległości 10, 60-161 Poznań
{witold.abramowicz,jacek.malyszko,dawid.weckowski}@kie.ue.poznan.pl
http://kie.ue.poznan.pl/

Abstract. Building and utilizing users' models to the end of better customized education products, especially in e-learning, is beneficial for both users and companies they work for. However, there is no general solution enhancing users' choices concerning their long–term learning goals. In this paper, we introduce a method, that trough analysis of time–variable user models provides a structured collection of information on common patterns of users' interests changes. Based on that, predictions can be made to reveal possible future changes in the fields of users' interests. We argue, that such predictions can be helpful in learning goals recommendation.

Keywords: lifelong learning, recommender systems, collaborative filtering, e-learning.

1 Introduction

Traditionally, learning paths were carefully planned by teachers, lecturers, deans [1] or, in organizations, by HR specialists[2]. Advent of ICT and e-learning created new opportunities in this area; personalization of learning paths became possible. Such personalization aims to make learning process more comprehensible and interesting to a learner, making the learning process more efficient [3].

eXtraSpec project proposed a solution to extract information about experts' skills from company's internal electronic documents and information available on the Internet. Each expert is represented in the system by its profile, which contains among other information references to concepts in a skill ontology. An enhanced indexing and searching mechanisms allow for efficient and precise retrieval of experts with certain competencies [4].

The motivation of the research presented in this paper is to utilize information about experts' competencies to plan their individual skills development paths. Moreover, such knowledge can be used for adjusting the structure of all trainings in organizations.

Basic elements of a learner's profile that are necessary for the personalization of learning paths are his current knowledge and his learning goal [5] . Such goal can be, for example, gaining skills in some programming language. These two

J. Grabis and M. Kirikova (Eds.): BIR 2011, LNBIP 90, pp. 231–238, 2011.

factors form an input to the learning paths personalization and therefore they must be known in advance [5].

At this point a question arises on how learning goals are determined. If, for example, HR specialists have to determine the structure of trainings in their organization, there can be two, most common, situations. In the first one, an employee wants to gain some new skills and a direction of his training must be determined. The other situation is when there are some skills missing in an organization and HR specialists must choose employees, which are to be trained to acquire these skills. In both situations, we have to either find a learning goal for a given person or find a person for a given goal.

In this article, we propose an extension to collaborative filtering methodology, that will enable utilization of information about other learners' interests evolution patterns in determining learning goals structure for learning paths personalization.

2 Related Work

2.1 Collaborative Filtering

Collaborative filtering (CF) is a method of content filtering, that is heavily used in recommender systems [6]. The main idea of this method is to programmatically augment the everyday process of making choices based on recommendations (either written or by word-of-mouth) from other people [7]. Computer systems can collect users' opinions on specified products and, by process of information aggregation or users matching, systems can predict user's rating of specific items and propose presumably the most interesting items for the user [6]. Depending on method of utilizing information about users and their preferences, two classes of algorithms can be pointed out: memory–based and model–based. Predictions in memory–based algorithms are calculated using all available information; initially a so-called neighborhood of a user or an item is assessed, and based on ratings assigned by (or given to) elements of such neighborhood, recommendations are formulated. Memory–based algorithms can be divided on user–based and item–based algorithms. The former employs user similarity measures to discover collection of appropriate recommenders [8], whereas item–based algorithms find items similar to those highly rated by the user, thus building a recommendation list [9].

Recently more and more attention is being paid to the other class of algorithms – a model–based approach. Such algorithms construct models of users' behavior, adjust parameters based on gathered information and then predictions are built by applying the model to a specific user. There are several approaches to build such models. Later in our article we extensively use clustering-based approach. Clustering-based methods are categorized as model–based, although the models often are built as a base for memory–based analysis, thus taking advantage from both approaches. Clustering analysis has long history in collaborative filtering adaptation and still new methods, including hybrid ones, are being developed [10].

In terms of lifelong learning, among other problems of collaborative filtering (and of recommender systems in general) it is worth to mention a so-called paradox of a profile information conservativeness [11], stating that the more precise recommendation for the user are, the less possibilities for development and broadening his knowledge he has. There are some efforts made to overcome that problem by including temporal analysis in prediction generation [12], or discovering new users' interests upon already identified ones [13]. We believe, that there is a need for constructing a more comprehensive description of user development paths and, especially in lifelong learning process, there is a need for recommendations concerning long–term perspective.

2.2 Collaborative Filtering for E-Learning

A traditional area of interest of collaborative filtering (CF) and recommender systems in general is e-commerce. Still, since the beginning of 2000s there have been many attempts to use this methodology for e-learning personalization. Probably the first article, that presents such approach, is paper [1]. In this article, an association rule mining algorithm was utilized to find rules, that may be used for learning object recommendation. Learners' access patterns are analyzed to extract frequent sequences of accessed URLs or actions in a following form: $I_1, I_2, I_3 \Rightarrow I_\alpha, I_\beta, I_\gamma$. If a learner is seen to follow an antecedent of such a rule, a consequent of the rule is recommended.

In the following years, many additional rules have been introduced to CF for e-learning. This was justified by the fact, that e-learning is much different from the traditional domain of CF utilization. In e-learning, recommendations cannot depend only on learners' tastes[14]; it should be determined not only by topic of a recommended article, but also by, among others, a level of knowledge that learner should have to understand it. Moreover, recommendations in e-learning have to comply to a goal of the learning process[5]. CF should therefore generate not a set of recommended articles, but a logical sequence of these, leading to a chosen goal.

An interesting example of approach facing this issue is utilization of ant colony algorithms. In these methods, during their education learners leave trails of learning paths, that they traversed; in ant colony systems, such trails are called pheromones. These pheromones can later be used to recommend learning paths for next learners aiming for the same goal[15]. Recommended learning paths are generated based on sequences of successfully completed learning actions. Such analysis allows to recommend to users several different paths; the fastest one, one with the highest level of learners' satisfaction etc. This method can be further extended to take into account the similarity of users by introducing different processing method of pheromones depending on learning style of users, that generated these pheromones [16].

3 Proposed Solution

3.1 Inputs and Outputs of the Proposed Method

The solution we present is designed to analyze changes, that occur in users' models over time in order to produce a graph representing their interests evolution patterns. Our basic assumption is that users' interests are not constant. If we take models of some user from several different months, they will probably be somewhat different; some interests will be more important, some less, some new will appear and some will completely disappear. System input is therefore a collection of users' models from different moments, that represent users' areas of interest in given periods. Such models from different periods we call *User Model States* (UMSs).

Areas of interest are represented using concepts from a defined ontology, that models knowledge domains. User is described by a subset of domain concepts (areas of his interests) with assigned weights (strength of his interest in this area). In eXtraSpec project, such model can be created for example based on certificates, that given user was recently awarded with. In experiments conducted so far, we chose a part of WordNet as our ontology – namely sub-concepts of *Knowledge Domain* concept.

As an output of the method a graph is generated, that represents users' interests evolution patterns. Vertices of the graph represent prototypes of clusters of UMSs (users' models states) . Graph edges indicate directions of user interests evolution. If there is an edge from vertex A to B, it means that at some moment there was a group of users, whose models were similar to the model represented by vertex A, and after some time some of those models changed in the way that could be generalized by vertex B. The weight of the edge informs about number of users taking this path of evolution. In every vertex a history is kept of all user models that have been ever associated with the given vertex in the past. Thanks to that, an analysis can be conducted on user models long–term evolution paths.

3.2 Description of the Algorithm

Let's assume, that a new adaptive system was introduced and it maintains users' models structured in the way described in section 3.1. Every three months, we make a snapshots of all users' models and we deliver these snapshots (which are *de facto* UMSs) as an input to our method. Our method works therefore in many iterations (in this case, one iteration every three months). The main tasks, that are performed in each iteration, are clustering (and, in later iterations, classification) of UMSs and generation of prototypes of created clusters. Let's discuss a few iterations to get the precise understanding on how our algorithm works. In the following sections we focus on the most important features of our approach; more detailed analysis of implementation of our method in terms of approach taken for clustering, similarity measures, generation of centroids etc. is beyond the scope of this paper.

First Iteration. During first iteration, initial clusters of UMSs are created; UMSs are divided here to a number of groups, based on similarity between them. Therefore, each cluster consists of some number of UMSs; it shows, that owners of these UMSs at a given point in time were interested in similar topics. These clusters will be necessary in the following iterations. In hitherto experiments, we used agglomerative hierarchical clustering approach.

Second Iteration. In second iteration, a system is updated with a collection of new snapshots. The basic idea here is to perform clustering of new snapshots together with snapshots from the first iteration to see, how interest's of users evolved since first iteration. If UMS of a given user from first iteration is in cluster C_A, and from second iteration in cluster C_B it means, that user changed areas of his interest and information can be inferred on the direction of such change.

To avoid processing again UMSs from the previous iterations, we introduced a modification to the method described in the paragraph above. At the end of previous iteration, for each cluster its prototype is generated; such prototype is an artificial UMS, that can be understood as a generalization (or centroid) of a given cluster. Additionally, in such prototype we put information about history of a cluster, that it represents. Such history is a list of UMS identifiers in a form of $UMS_{X,Y}$, where X is a user ID and Y is a number of iteration, in which given UMS was recorded in this cluster. An example of such history is presented in a form of a table on figure 1. In this and next iterations, in clustering phase we use such prototypes instead of all UMSs from the previous iterations to avoid performing the same computations multiple times.

After clustering phase, in second iteration new clusters were generated. Some of these new clusters contain prototypes of clusters from the first iteration. Now, we generate prototypes for the new clusters. During generation of the new artificial UMSs, that will be the centroids (prototypes) of new clusters, we assign a higher weight to the prototypes from a previous iteration based on how many UMSs are recorded in their history, so that the new centroid is closer to the one form the previous iteration. Next, we copy a history records from old prototypes to the new ones and we update the history of new prototypes with records about UMSs, that in second iteration were assigned to clusters, that these prototypes represent. Finally, we remove prototypes from the previous iteration from the system.

Next Iterations. Every three months, the system is incrementally updated with new snapshots. After each iteration, prototypes from the previous iteration are removed and a new collection of prototypes is generated. History records of prototypes from the previous iteration are copied to prototypes of the new clusters, that these old prototypes are assigned to. With each iteration, positions of the prototypes should stabilize (in other words, position of the new prototypes should be very similar to the old ones, that they contain), as history record is growing and more and more weight is put on prototypes from the previous iteration. At a certain moment, for example when the history size of a given

Prototype A history records	
Iteration no	User IDs
1	1,5,10,**12**
2	3,**8**,5
3	**2**,7,11,13

Prototype B history records	
Iteration no	User IDs
1	4,15
2	**2**,**12**,18
3	**8**,15,16

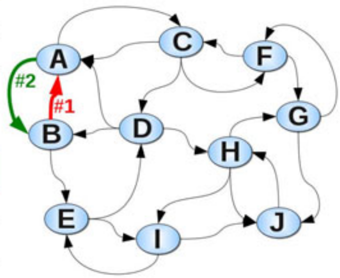


Fig. 1. History records for two prototypes and a resulting graph. There were two transitions between node A and B (user 12 and user 8) and one between node B and A (user 2).

prototype exceeds a certain number of records or when a distance between the new prototype and the old one is smaller than a chosen threshold, a decision can be made to freeze the prototype to make sure, that it doesn't change it's position anymore.

We treat such frozen prototypes in a special way. In the subsequent iterations, before clustering phase we perform a classification of new UMSs to the frozen prototypes; for each frozen prototype, we update it's history with all UMSs, that are closer to it than a chosen threshold, and we remove these UMSs from further processing in this iteration. On remaining UMSs and prototypes, that are not frozen yet, we perform a regular clustering as described before.

Therefore, after every iteration a collection of prototypes with history records is generated. By analyzing these history records, we can discover changes in areas of users' interests.

3.3 Interpretation of the Output

To make results of our method easier to analyze for prediction purposes, history records are transformed into a so-called transition table $T_{i,j,[m,u]}$, which shows individual changes of user models (m) classification from cluster i (row name) to cluster j (column name) observed on update u (diagonal elements mean, that interests of users didn't change between two iterations). Example of a transition table is presented on table 1.

Simple analysis of a transition table can indicate directions of users' models evolution. By rows examination one can discover outgoing paths, indicating how interests of a users, that in some iteration were assigned to the prototype represented by a given row, changed in the following iterations. For example, element $[1, 2]$ in cell c2/c3 on table 1 means, that in second iteration it was observed, that interests of a user with id 1 in previous iteration were similar to the prototype $c2$

Table 1. Transition table

	c1	c2	c3
c1	[2,1] [2,2]	[1,1] [3,2]	[2,3]
c2	[1,5]	[2,5]	[1,2] [3,3]
c3		[1,4] [2,4] [3,5]	[1,3] [3,4]

and now are more similar to $c3$. On the other hand, columns analysis can reveal inbound traffic – what are the main prototypes, that users categorized in some iteration to a given prototype came from.

Concerning user recommendations, the most interesting aspect are long–term model evolution patterns and similarity of such patterns (their backward compatibility) between multiple different users. We can analyze conformity of several transitions to enhance prediction formularization. Thus if a user model evolution follows a longer path, that have been taken earlier by many other users, it is more likely that his further evolution will stay on that path rather than switch to another one. Taking Table 1 as an example, although model 3 in update 5 made the same transition as models 1 and 2 in update 4, there is a higher probability that in the sixth update model 3 will make the same transition as model 1 in update 5, due to consistent history of evolution of both models.

4 Conclusions and Future Work

Presented method allows us to predict possible future areas of interests of a given user. We argue, that such knowledge can be used to determine learning goals in lifelong learning. For example, a HR specialist that has to assign learning goals to employees can check, what employees are probably going to be interested in in the future and compare it with a list of missing skills in the organization. Assigning users to trainings, that comply with their predicted future evolution patterns can probably make the trainings more effective and improve employees satisfaction.

An interesting extension to described methodology would be to develop a method of automatic matching of predictions of future users' interests evolution paths with a list of missing skills in the organization, to make the whole process even more effective. Future research directions should include also further experimentation on real-life data.

Acknowledgements. The work published in this article was supported by the project titled: "Advanced data extraction methods for the needs of expert search" (http://extraspec.kie.ue.poznan.pl) financed under the Operational Program Innovative Economy and partially supported by European Union's European Regional Development Fund programme (contract no. UDA-POIG.01. 03.01-30-150/08-01)

References

1. Zaíane, O.R.: Building a recommender agent for e-learning systems. In: Proceedings of the International Conference on Computers in Education, ICCE 2002, pp. 55–59. IEEE Computer Society, Washington, DC (2002)
2. Kelly, D.: Dual Perceptions of HRD: Issues for Policy: SME's, Other Constituencies, and the Contested Definitions of Human Resource Development (2001)
3. Sterbini, A., Temperini, M.: Adaptive construction and delivery of web-based learning paths. In: Proceedings of the 39th IEEE International Conference on Frontiers in Education Conference, FIE 2009, pp. 1223–1228. IEEE Press, Piscataway (2009)
4. Abramowicz, W., Bukowska, E., Kaczmarek, M., Starzecka, M.: Semantic-enabled efficient and scalable retrieval of experts. In: International Conference on Information, Process, and Knowledge Management, eKNOW 2011 (2011)
5. Drachsler, H., Hummel, H., Koper, R.: Recommendations for learners are different: Applying memory-based recommender system techniques to lifelong learning. In: Proceedings of the EC-TEL Conference, Crete, Greece, Citeseer (2007)
6. Cacheda, F., Carneiro, V., Fernández, D., Formoso, V.: Comparison of collaborative filtering algorithms: Limitations of current techniques and proposals for scalable, high-performance recommender systems. ACM Trans. Web 5, 2:1–2:33 (2011)
7. Resnick, P., Varian, H.R.: Recommender systems. Commun. ACM 40, 56–58 (1997)
8. Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., Riedl, J.: Grouplens: an open architecture for collaborative filtering of netnews. In: Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work, CSCW 1994, pp. 175–186. ACM, New York (1994)
9. Sarwar, B., Karypis, G., Konstan, J., Reidl, J.: Item-based collaborative filtering recommendation algorithms. In: Proceedings of the 10th International Conference on World Wide Web, WWW 2001, pp. 285–295. ACM, New York (2001)
10. Treerattanapitak, K., Jaruskulchai, C.: Membership enhancement with exponential fuzzy clustering for collaborative filtering. In: Wong, K.W., Mendis, B.S.U., Bouzerdoum, A. (eds.) ICONIP 2010, Part I. LNCS, vol. 6443, pp. 559–566. Springer, Heidelberg (2010)
11. Abramowicz, W.: Filtrowanie Informacji. Wydawnictwo Akademii Ekonomicznej w Poznaniu (2008)
12. Liu, N.N., Zhao, M., Xiang, E., Yang, Q.: Online evolutionary collaborative filtering. In: Proceedings of the Fourth ACM Conference on Recommender Systems, RecSys 2010, pp. 95–102. ACM, New York (2010)
13. Hijikata, Y., Shimizu, T., Nishida, S.: Discovery-oriented collaborative filtering for improving user satisfaction. In: Proceedings of the 14th International Conference on Intelligent User Interfaces, IUI 2009, pp. 67–76. ACM, New York (2009)
14. Santos, O., Boticario, J., Baldiris, S., Moreno, G., Huerva, D., Fabregat, R.: Recommender Systems for Lifelong Learning inclusive scenarios. In: Workshop on Recommender Systems, Citeseer, Citeseer, p. 45 (2008)
15. Janssen, J., Tattersall, C., Waterink, W., van den Berg, B., van Es, R., Bolman, C., Koper, R.: Self-organising navigational support in lifelong learning: How predecessors can lead the way. Comput. Educ. 49, 781–793 (2007)
16. Wang, T.I., Wang, K.T., Huang, Y.M.: Using a style-based ant colony system for adaptive learning. Expert Systems with Applications 34(4), 2449–2464 (2008)

Modeling Secure Navigation in Web Information Systems*

Marianne Busch[1], Alexander Knapp[2], and Nora Koch[1,3]

[1] Ludwig-Maximilians-Universität München,
Oettingenstraße 67, 80538 München, Germany
{busch,kochn}@pst.ifi.lmu.de
[2] Universität Augsburg,
Universitätsstraße 6a, 86159 Augsburg, Germany
knapp@informatik.uni-augsburg.de
[3] Cirquent GmbH,
Zamdorfer Staße 120, 81677 München, Germany

Summary. Secure web information systems are becoming increasingly important due to rising cybercrime as well as the growing awareness of data privacy. Besides authentication and confidential connections, both data access control and navigational access control are the most relevant security features in this field. Adding such security features, however, to already implemented web applications is an error-prone task. Our approach enables web engineers to model security issues in an early phase of the development process. We demonstrate the integration for the UML-based Web Engineering (UWE) method. The approach supports the engineer by providing means to model navigational security with a plugin in a UML modeling tool. Additionally, the models can be used for the verification of web systems and security properties, such as reachability of navigation nodes in general and of those that are restricted to authorized users.

Keywords: Security, Web Engineering, Modeling, Verification.

1 Introduction

The article *Top 25 Most Dangerous Software Errors*[1] clearly shows the relevance of security aspects in software systems. The list includes items like "Improper Access Control (Authorization)", "Missing Encryption of Sensitive Data" or "Missing Authentication for Critical Function". These threats are exacerbated by the global and 7/24 accessibility of web information systems (WISs) as well as the unforeseeable range of customers. Therefore many of those systems require role dependent sessions. Login mechanisms mostly imply not only changed

* This work has been partially supported by the DFG project MAEWA II, WI 841/7-2, the EU project ASCENS, 257414, and by the EU-NoE project NESSoS, 256980.

[1] Common Weakness Enumeration CWE/SANS. Top 25 Most Dangerous Software Errors. http://cwe.mitre.org/top25/#Details, last visited 2011-02-18.

J. Grabis and M. Kirikova (Eds.): BIR 2011, LNBIP 90, pp. 239–253, 2011.

permissions regarding accessible data or activities, but also access to particular non-public areas of a website. Restricted access often goes hand in hand with the need to take care of freshness, confidentiality and integrity while transmitting data over an insecure network as the Internet.

More often than not security is added to already implemented software, i.e. too late in the development process, for example after data leaks have been detected. However, incidents like publicly accessible credit card details or personal registration data could be mostly avoided using an engineering method supporting the security concern from the beginning of the development process. In fact, web application frameworks, such as Spring, Joomla, RubyOnRails, Lift and Zope[2] offer support for the implementation of security aspects. The support varies from integrated elements in the language to specific modules or plugins that need to be explicitly installed, mainly based on access control lists (ACL). What still is missing is the possibility to handle these security topics earlier in the web application life cycle.

Our aim is to provide convenient modeling techniques that enable WIS developers not only to figure out what customers need exactly, but also how their idea of security can be specified in a concrete and intuitive manner and afterwards seamlessly implemented in any selected framework. In particular, our approach allows the developer to model security features such as access control, authentication and secure connections graphically with UML. The aim of authentication is to gain access to a protected resource. The process of authorization determines what a *subject* (e.g., a user or a program) is allowed to access, especially what it can do with specific *objects* (e.g., files) [1]. First and foremost, WISs have to protect not only data and restrict which functions can be called, but also have to take navigational access control into account, i.e. the parts of webpages that are accessible by a certain user. If only a certain user should have access, it is likely that the connection should be secured, i.e. confidentiality, integrity and freshness of the transmitted data have to be ensured. This guarantees that the data cannot be eavesdropped. In addition, replayed or altered information is recognized immediately.

Our UML extension for security aspects is a class and statechart-based approach that uses these techniques for a model-driven development of secure web information systems. Class diagrams are used to specify the content as well as the rights model, statecharts yield precise UML-based navigation models. Additionally, the statecharts can be subjected to model checking for verifying reachability of navigation nodes in general and of those that are restricted to authorized users. In fact, UWE's security features could be used in combination with any UML-based web engineering approach. The web engineer can develop secure web information systems in his favorite UML CASE tool; he only needs to include the UML-based Web Engineering (UWE) profile which extends the UML with a set of web and security features. Additional support is provided by

[2] Wikipedia: *Comparison of web application frameworks.* `http://en.wikipedia.org/wiki/Comparison_of_Web_application_frameworks`, last visited 2011-05-15.

the MagicUWE plugin for MagicDraw[3], which eases the modeling task by a set of direct accessible stereotyped elements, shortcuts and patterns.

The remainder of this paper is structured as follows: Section 2 outlines the main characteristics of web engineering approaches, in particular their implementation in UWE, and links them to security. Section 3 presents the UWE security extension focusing on navigation and data access control, where we use a simple online address book for illustration. How to work with our approach is described in Sect. 4, presenting the tool support for constructing models and their use for verification. Section 5 discusses related work. Finally, we give an outlook on future steps in the development of secure web information systems.

2 Web Engineering

Web Engineering is a specific domain in which model-based software development has been successfully applied. Existing approaches, such as OOHRIA [14], OOWS [18], UWE [12], or WebML [16] already provide well-known methods and tools for the design and development of web applications. Most of them follow the principle of "separation of concerns" using separate models for views, such as content, navigation, presentation, business processes, et cetera.

Content. The content model is used to represent the domain concepts that are relevant for the web application to be built and the relationships between them. Visualization of certain content is very often associated to a successful authentication, i.e. require a name and a predefined, correct *password* or a *digital certificate.*

User Model. A user or context model can be used to collect information needed for adaptation. A role model is a special case of user model, in which characteristics of the user groups are defined with the purpose of authorization and access control. Very often content and user model are integrated.

Navigation. The navigation model is used to represent navigable nodes of the hypertext structure and the links between nodes. The existing approaches differ in the representation of these web-specific concepts. Some see the nodes as pages of WISs; others distinguish between the idea of navigation node and page, where a page can be composed of several navigation nodes. There is also a difference in the choice of a structural or behavioral representation for the navigation structure. Navigational nodes are constrained by navigational access control rules, which means that users without permission can see only an error message or an advertisement for a less restrictive account. Furthermore, information accessed through navigation may have specific requirements on the confidentiality, integrity and freshness of data of the web application content.

Presentation. There are two trends in the specification of layout concerns in web engineering. On one hand the use of prototypes, on the other hand sketching the user interface using mock-up tools or modeling the presentational aspects. The objective is either a detailed platform-specific specification of the user interface

[3] MagicDraw. `http://www.magicdraw.com`, last visited 2011-02-20.

or the rough layout of the pages. In any case not only the static GUI widgets but also features of Rich Internet Applications (RIAs) like auto-completion in search fields, live validation of input fields, or drag & drop functionality should be represented in the presentation model. Regarding security features privacy also plays a role for the adaption of presentational aspects, e.g., private calendar entries should not be shown during working meetings.

Process. The process model aims to represent the workflows which are invoked from certain navigation nodes. The same security considerations apply as for the navigation model.

UWE strongly supports this "separation of concerns" selecting the appropriate UML diagram type and elements for each web concern and using the UML extension mechanisms, i.e. defining a set of stereotypes and tagged values in a profile. Concepts of the content and user model and their relationships are shown as classes and associations in a UML class diagram. For the navigation model UWE provides two different graphical representations: a structural visualization as UML stereotyped class diagrams and a behavioral form using UML state machines, which eases the specification of security features. UWE's presentation model is visualized like a mockup, using composite structure diagrams in which composition is visualized as nested classes and properties. Stereotypes are used for GUI widgets, and tagged values to represent RIA functionality. The process model comprises two views: first the process structure model that describes the relations between the different process classes, which are related to the navigation, and second the process flow model that shows the workflow for each process. They are represented by UML class diagrams and UML activity diagrams, respectively. In the following, we add to UWE a model for access control and extend UWE's behavioral representation of the navigation model.

3 Modeling Access Control in Web Information Systems

We integrate the modeling of access control, both for data and navigational access, into the modeling of WISs and RIAs. This integration enables the modeler to address security aspects right from the early phases of the development life cycle. We rely on standard UML modeling techniques and the definition of a UML profile extending UWE. The profile enhances class diagrams to specify a basic rights model building on role-based access control (RBAC) and state machines for modeling controlled navigation. We illustrate our approach by a secure address book application.[4] Due to lack of space, not all features of our security extension are used in this example, for further information the reader is referred to [3].

Case study. The WIS should allow registered users to create and navigate several address books and to add and retrieve contacts in them. Non-registered visitors can only read an introduction and the terms of service until they register or

[4] More information about the secure address book case study can be found at `http://uwe.pst.ifi.lmu.de/exampleSecureAddressBook.html`, last visited 2011-05-07.

authenticate themselves. Administrators cannot use the address book functionality, but they are allowed to search for users and to delete their accounts including all address books and contacts.

3.1 Basic Rights Model

The basic rights model is used to specify access control rules for domain concepts which are represented as UML classes and class instances. The Role instances from UWE's user model and content (or user model) classes, their attributes, and methods are connected with stereotyped dependencies. These dependencies, on the one hand, specify create/read/update/delete (CRUD)-rights; on the other hand, an execution dependency between a role and a method grants execution rights for the method.

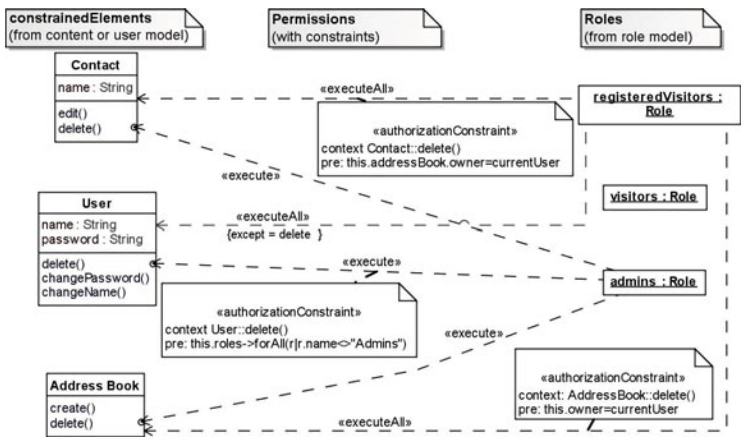


Fig. 1. Address book: Basic rights model

For the address book example, the basic rights diagram in Fig. 1 specifies execution rights on methods with dependencies stereotyped ≪execute≫ and ≪executeAll≫ (the CRUD support using ≪read≫/≪readAll≫ etc. is not shown in this example). The dependencies connect the role instances shown on the right in Fig. 1 to the methods of the content model, like Contact and AddressBook, or of the user model, such as the class User, on the left. In particular, a non-registered visitor has no execution permissions. The {except} tag for ≪...All≫ stereotypes allows the modeler to avoid the creation of too many dependencies. For instance, a registered visitor can execute all methods of a user object except delete. Further restrictions are added in comments stereotyped by ≪authorizationConstraint≫ in the Object Constraint Language (OCL): A registered visitor shall only be allowed to delete his own contacts and address books; an administrator shall have the permission to delete all users except other administrators. The corresponding restrictions on ≪execute≫ for Contact::delete(), AddressBook::delete(), and

User::delete() use attributes like AddressBook.owner and User.roles from the content model. The pre-defined currentUser refers to the user of the current session.

UWE's basic rights model offers a compact notation for access specifications where permissions and prohibitions can be readily read off. This is in contrast to approaches like SecureUML [13] where all permissions have to be specified separately in association classes, and exceptions cannot be expressed. However, transformations between SecureUML and our basic rights model are possible.

3.2 Navigation State Model

A navigation state model describes the navigation structure of a WIS and its behavior according to the different states. In UWE, navigation can be represented by a UML state machine: States, possibly hierarchical, represent navigational nodes, transitions the navigational links between the nodes. The UWE security profile allows to integrate navigational access control, but also session management and secure connections into the state machines specifying navigation. In particular, the navigational state model should be aligned with the access control mechanisms in the basic rights model, as e.g., a user who is not allowed to access a function of a class should be disallowed to navigate to a node that uses this functionality and vice versa.

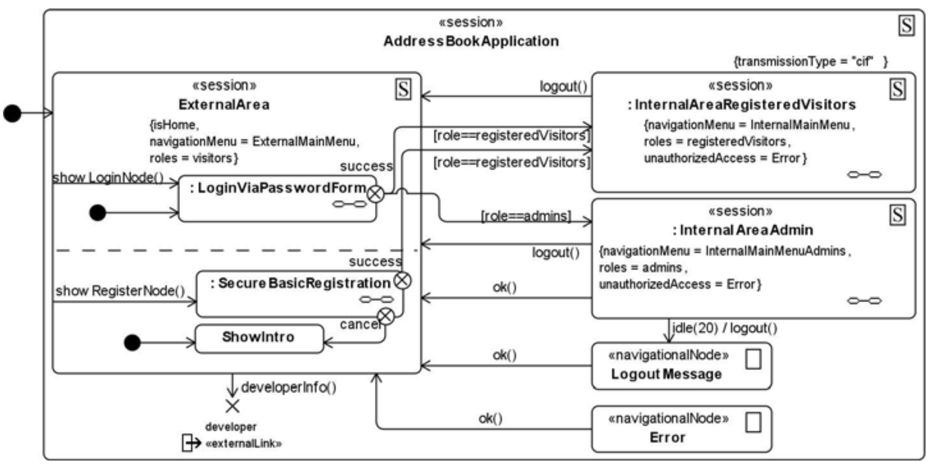


Fig. 2. Address book: Outermost navigation state model

Figure 2 depicts the main navigation state diagram for the address book example. All states are navigational nodes; when the system is in a particular state, the information and behavior offered by this state is accessible.

Navigation starts inside ExternalArea (pointed to from the outermost initial state) which is a substate of AddressBookApplication. Here, the substate machine LoginViaPasswordForm (indicated by ⚯) and state ShowIntro are entered simultaneously (as ExternalArea shows two regions separated by a dashed line). ExternalArea is the starting point of the WIS, tagged by {isHome}. The tag {navigationMenu=ExternalMainMenu} tells that when inside ExternalArea and whichever substates, the user can access the actions from the navigation menu ExternalMainMenu, which include showLoginNode, showRegisterNode, and developerInfo (we omit a UML representation as a class diagram). When showLoginNode is selected, LoginViaPasswordForm is entered, when showRegistrationNode is chosen SecureBasicRegistration is entered. Both substate machines are, in fact, instances of the UWE security patterns offered for recurring security issues to be discussed in Sect. 4.1 (see also Fig. 4 to the left and Fig. 5).

After successful login or successful registration (leaving the two success exit points), two types of internal areas can be reached: one for the administrators and one for the registered users who want to manage their contacts. However, the subsequent area depends on the role from the role model the user takes on during login (we assume that role visitors is the default role): The guards on the transitions targeting the internal areas check the access rights. In these guards, currentUser.role is abbreviated to role.

The areas ExternalArea, InternalAreaRegisteredVisitors, and InternalAreaAdmin, as well as the super-state AddressBookApplication are distinguished navigational nodes: Their stereotype ≪session≫ (Ⓢ) shows that context information on the navigating user is kept. The areas also restrict navigational access to them to particular roles by using the tag {roles=...}, such that, e.g., InternalAreaRegisteredVisitors can only be entered by registeredVisitors. This restriction not only protects against access through navigational transitions which should show appropriate guards, but also prohibits direct unauthorized access via a URL. Additionally, the tag {unauthorizedAccess=...} specifies which state is entered when the access rule is violated; for both internal areas this state is Error. The tag transmissionType="cif" for the session state AddressBookApplication sets the overall type of data transmission during the session to cif, providing for confidentiality, integrity, and freshness: The implementation should prevent eavesdropping, replaying, or altering of transmitted data. The transmission type is sustained also in the substates.

InternalAreaAdmin can be left explicitly by choosing logout from its navigation menu InternalMainMenuAdmins; it will also be left when the user stays idle for more than 20 time units after which the application will transit into the navigational node LogoutMessage. Finally, the ≪externalLink≫ (⇨) developer can be reached from ExternalArea. When this external web page is opened in a new browser window or tab, the system will still be inside ExternalArea, otherwise the WIS is left.

UWE navigational states profile. The excerpt of the UWE profile in Fig. 3 summarizes the integration of navigation and security we have illustrated for the address book example.

The basic state and state machine stereotype for navigational state models is ≪navigationalNode≫. Here, "navigational" refers to the view and the granularity of the state machines, because not all states and transitions need to

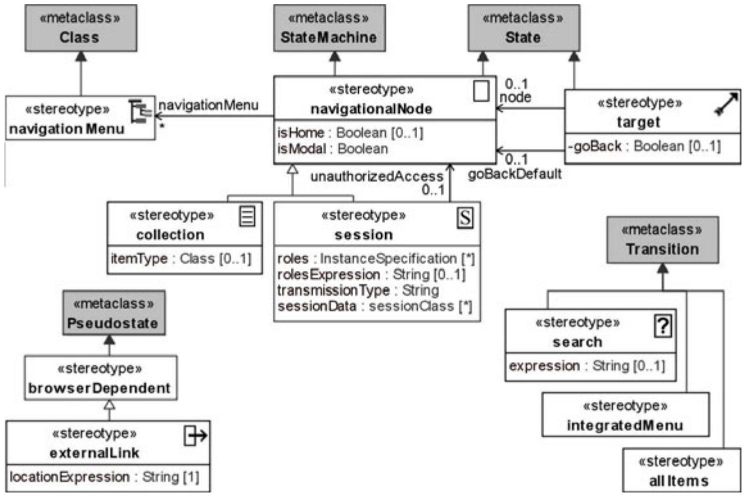


Fig. 3. Navigation states profile

represent navigational behavior. In particular, an ≪externalLink≫ is not a proper navigational state. The initial node of a WIS is marked by isHome. A ≪navigationalNode≫ can also be set as isModal, meaning that no other navigational node of the navigation model can be accessed as long as the modal node is active. Each ≪navigationalNode≫ can also refer to ≪navigationMenu≫s containing the operations which can be chosen from within the node. The ≪session≫ stereotype is derived from ≪navigationalNode≫ and thus inherits navigationMenu and isHome; a session additionally keeps sessionData and specifies role access restrictions (roles, unauthorizedAccess, and rolesExpression for more fine-grained rules) and a transmission type.

The UWE profile offers some further features for modeling navigation (cf. [3]): The tag {goBack} of the stereotype ≪target≫ (↗) allows to navigate to the state which previously had been active. Access to collections, e.g., to lists, is specially supported. Also, large menus can be integrated on transitions, which is particularly useful if each user can be associated with a set of roles.

Transformation to code. The UWE profile for integrating navigation and security in WIS is constructed in a way that transformation to code can be achieved in the near future. Such a transformation can utilize navigational access control offered by several web frameworks. The states of the navigation state model become HTML-fragments and the annotated access rules — together with UWE's role model — user role representations and rules for the web framework. For instance, the Scala-based Lift framework[5] controls navigational access via its site-map feature: A list of menu items connects all HTML fragments with the modeled access rules. Here, for InternalAreaRegisteredVistors a rule has to be stated that

[5] Lift. `http://www.liftweb.net/`, last visited 2011-05-01

the current user has to be associated with the role registeredVisitors. This is specified as an immutable variable in Scala and is used to regulate the access to every HTML fragment within this internal area.

Representation of Navigational Stereotypes in Plain UML. The precise meaning of UWE's navigational and security stereotypes and tags is defined by transformations into plain UML. Here, we only illustrate one such transformation for the running address book example; all transformation definitions can be found in [3]. In particular, the extended plain UML model resulting from applying the transformations can be subject to verification by model checking, as presented in the next section.

We translate the session tags {roles=admins} and {unauthorizedAccess=Error} for InternalAreaAdmin (see Fig. 2) into UML. When removing the tags, we have to ensure that no (sub)state of InternalAreaAdmin can be accessed without showing role admins, either directly or by recording a URL, but these users have to be redirected to Error. InternalAreaAdmin has two substates for which a URL could be recorded, one for searching a user and another for deleting a user (we omit the internal state machine of InternalAreaAdmin). Figure 4 shows the result of the transformation for this particular situation.

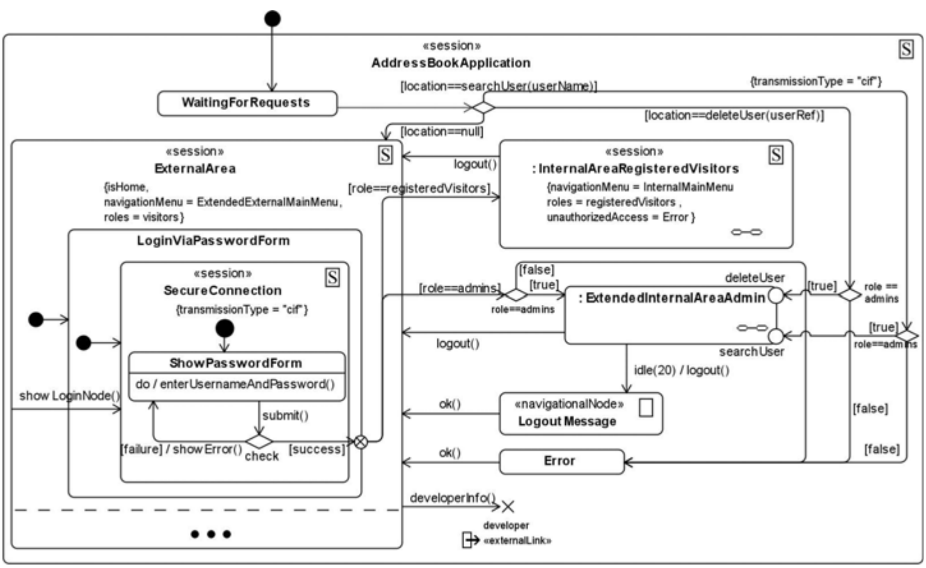


Fig. 4. Address book. Partly extended navigation state model.

If a user stores a URL, which is modeled as location variable that is set while WaitingForRequests, but is not logged in or is no longer allowed to take on the admin role, the application navigates to the Error state. (1) The two choices on the right have equal conditions, but guard the admin area from direct access.

(2) The choice in the center of the diagram protects the admin area from unauthorized access from ExternalArea (however, this is not necessary, because the guard on the transition before is strong enough). On the left of the extended state machine in Fig. 4, we have also unfolded the login pattern and we hide the registration functionality by '. . .'.

4 Working with UWE Security Models

The aim of our approach is to allow web engineers to address security aspects in an early phase of the development process. For this purpose the UWE profile includes modeling elements and diagrams specific to the web and security domain, providing a so-called domain specific modeling language (DSML). However, to further ease the web engineer's work a pattern catalogue for recurring issues and tool support for modeling and model validation are needed as well. Therefore, we added navigational patterns to the UWE profile, implemented a series of modeling supporting features as part of a plugin for the UML CASE tool MagicDraw, and offer the possibility to formally check model properties.

The applicability of the overall approach is proven by the design and implementation of a real (although simplified in the sense of focusing on security aspects) web-based hospital information system (HIS). For a detailed description and the sources the reader is referred to the web page of HospInfo[6].

4.1 Security Patterns

Patterns are a common approach to tackle the problem of repetitive tasks. "A security pattern describes a particular recurring security problem that arises in a specific security context and presents a well-proven generic scheme for a security solution." [17] We use security patterns specified as state machines that can be easily included as substate machines in navigation state diagrams. Typical examples are registration, authentication (login mechanisms), credential recovery (lost password), or profile configuration. We only explain the registration pattern, further examples can be found in [3].

For user registration commonly at least two things have to be checked: The user to be registered should be human; and the information the user provides has to be valid, for example the given email address should have the correct format. Another frequent requirement is to encrypt the entered data during the transmission to the server. Accordingly, the registration pattern, see Fig. 5, is modeled as a substate machine stereotyped as session and comprising a session state representing the secure connection by the tag {transmissionType="cif"}. Inside SecureConnection, a CAPTCHA[7] to tell computers apart from humans and the input of user data are offered. Only if both regions have been filled in

[6] HospInfo. A secure hospital information system `http://uwe.pst.ifi.lmu.de/exampleHospInfo.html`, last visited 2011-04-20.

[7] An example is Google's reCAPTCHA. `http://www.google.com/recaptcha`, last visited 2011-02-10.

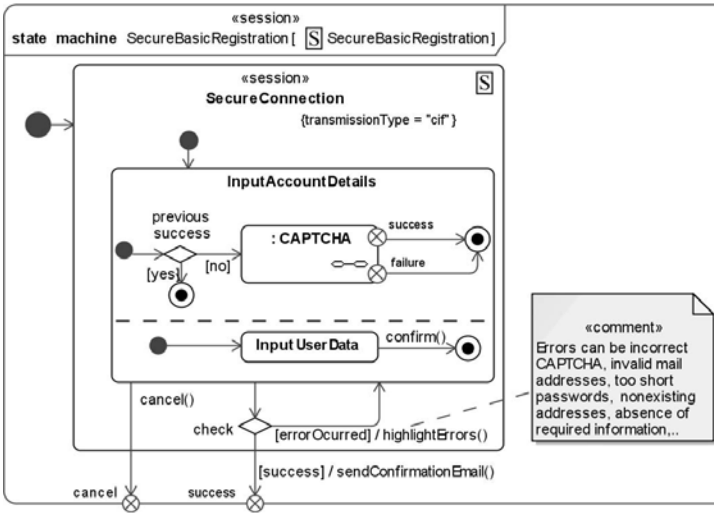


Fig. 5. Secure registration pattern

and no errors are detected by the check registration is successful. In fact, this check as well as the input data to be provided have to be customized when the pattern is applied.

4.2 Tool Support for Modeling

Tool support is crucial for the applicability of a methodology such as UWE. We decided to rely on the MagicDraw CASE tool for modeling all kinds of web applications [4]. Nevertheless, our tool concept may be adopted for other commercial and open source UML CASE tools.

MagicUWE[8] is a MagicDraw plugin for developing secure WISs with the UWE UML-profile in order to ease the modeling activities. Whenever models are created, some tasks have to be repeated over and over. Furthermore, some consistency checks and transformations are very time consuming, if executed manually. The solution is to provide plugin features that provide shortcuts for tasks like (1) inserting UWE's stereotyped elements directly from the toolbar and (2) copying UWE stereotypes and their tags between a state machine and its substate machines, (3) specifying tags facilitated by a context menu, (4) deriving the type of a substate from the stereotypes of a superstate recursively and (5) inheriting stereotypes for use cases stored in a package and (6) checking features of the models. Figure 6 depicts some of these plugin features.

An example for such a functionality is the check whether the transmission type of a secure connection is changed within nested states of the application's navigation model. This allows the modeler to see, if a service as e.g. a CAPTCHA

[8] MagicUWE. `http://uwe.pst.ifi.lmu.de/toolMagicUWE.html`, last visited 2011-05-08.

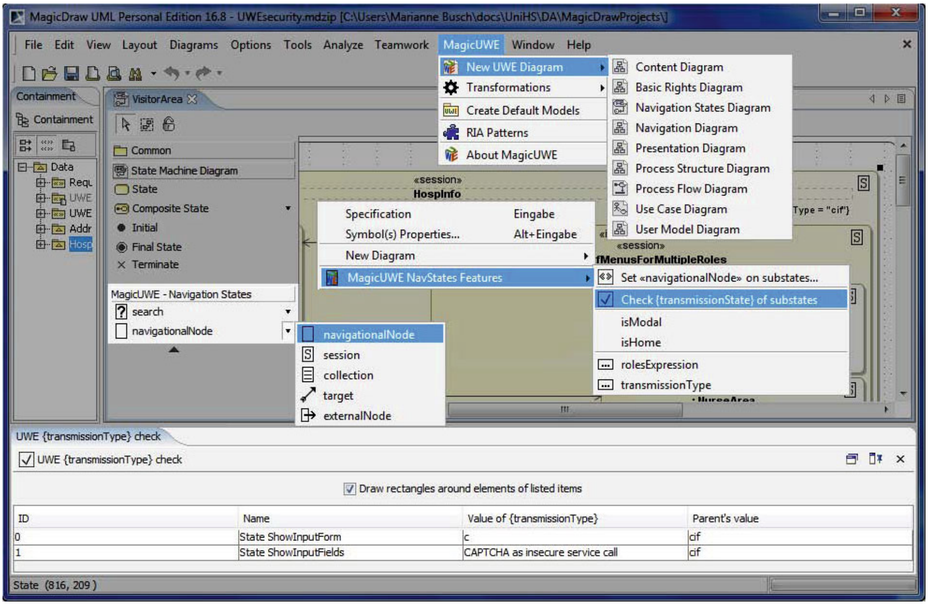


Fig. 6. MagicUWE plugin

that is used within a secure area communicates over an unencrypted connection. Without MagicUWE, all substates would have to be checked for changes by hand, which can be very time-consuming for larger models.

4.3 Validation of Models

The expressiveness and flexibility of the UWE profile and security modeling techniques makes it desirable to obtain feedback whether the model indeed satisfies the modeler's security intentions. Since UWE builds on standard UML and, in particular, UML state machines, by reducing many security modeling features to plain UML expansions, we may apply formal techniques developed for standard UML, like model checking [11,7] or theorem proving [2].

We use the UML model checking tool Hugo/RT [11] to check the UML state machine in Fig. 4, which results from the UWE navigation model, for security problems. For example, we want to ensure that no unauthorized user can enter the state ExtendedInternalAreaAdmin, i.e., whenever in this state the current user must play the role admins. In the temporal OCL extension supported by Hugo/RT this property reads:

```
G a.inState(AddressBookApplication.
              ExtendedInternalAreaAdmin) implies
    a.currentUser.role == ADMINS;
```

where G is the linear-temporal logic operator "always" and object a represents the application.

The WIS has little control on what a user may try in order to reach some location, be it that the user just follows links or that the user browses to a location directly using link guessing or previously recorded links. We thus build (currently manually) an attacking user who tries all possible interactions with the web application in all possible ways; this user is again represented as a UML state machine. Hugo/RT translates the state machines for the web application and the user, as well as the assertion into the input language of a back-end model checker, in this case SPIN [9]. SPIN then verifies that the assertion indeed holds. In fact, such a property may look quite obvious in our running example; however, the situation can get rather complicated in bigger applications, like HospInfo.

5 Related Work

This work is related to several security and web engineering approaches, mainly to those which focus on UML-based specification of secure systems.

UMLsec [10] is an extension of UML with emphasis on secure protocols. It is defined in form of a UML profile including stereotypes for concepts like authenticity, freshness, secrecy and integrity, role-based access control, guarded access, fair exchange, and secure information flow. In particular, the use of constraints gives criteria to evaluate the security aspects of a system design, by referring to a formal semantics of a simplified fragment of UML. UMLsec models compared to UWE models are very detailed and therefore quickly become very complex. The main difficulty is the missing support of UML 2 since the provided UMLsec tools[9] and the used CASE tool ArgoUML[10] still only supports UML 1.4.

SecureUML [13] is a modeling language for the model-driven development of secure, distributed systems also based on UML. It provides modeling elements for role-based access control and the specification of authorization constraints. A SecureUML dialect has to be defined in order to connect a system design modeling language as, e.g., ComponentUML to the SecureUML metamodel, which is needed for the specification of all possible actions on the predefined resources. In our approach, we specify role-based execution rights to methods in a basic rights model using dependencies instead of the SecureUML association classes, which avoids the use of method names with an access related return type. However, UWE's basic rights models can easily be transformed into a SecureUML representation.

There is a set of approaches that address modeling of security aspects of service-oriented architectures (SOAs), such as the SECTET framework [8], the SENSORIA approach UML4SOA [6], and SecureSOA [15]. The first one proposes the use of sequence diagrams for the representation of a set of security patterns, in UML4SOA security features are modeled as non-functional properties using class diagrams, and the latter relies on FMC block diagrams and BPMN notation.

[9] UMLsec Analysis Tools. `http://ls14-www.cs.tu-dortmund.de/main2/jj/umlsectool/`, last visited 2011-03-15.

[10] ArgoUML. `http://argouml.tigris.org/`, last visited 2011-03-20.

6 Conclusions and Future Work

We have addressed access control and other security features in an early phase of the development process of web information systems. Modeling elements and patterns for RBAC, secure communication links and authentication-related processes are provided, which can be applied to navigational aspects of the web information system. UWE's security features are defined in such a way that repetitions are avoided whenever possible and security-specific modeling elements are offered according to the granularity needed for the implementation. UWE is a UML-based approach based on state machines for the representation of secure navigation and secure patterns. The UWE profile is UML-compliant, i.e., usable by any UML CASE tool. Additional comfort in the modeling process is provided by the MagicDraw plugin [4].

The applicability of our approach is proven by two case studies: a hospital information system and a secure address book. The first one covers the full WIS development life cycle, from requirements analysis, through modeling to a concrete implementation in Scala using the Lift framework. The second one was used for illustrating the advantages of a concise notation and the verification facilities of the navigation state machines introduced in this work that are checked using the Hugo/RT UML verification tool.

Future work will encompass further validation consisting in the combination of UWE's security features and other web engineering methods, such as the UML version of WebML [16]. In addition, we plan to extend the current UWE approach to cover the generation of a database scheme for the access rights. For this purpose we will consider the ongoing research work of Egea et al. [5] concerning SecureUML [13] and databases. The corresponding transformations will be implemented in our MagicUWE plugin. Another interesting issue is the separation of concerns regarding security aspects (e.g. as a list of requirements) and our navigation states model. A combination with the aspect-oriented modeling approach HiLa from Zhang et al. [19] might be promising. Furthermore, we are working on a code generator that transforms UWE navigation models to Scala, especially the page structure and the according access rights.

References

1. Anderson, R.J.: Security Engineering: A Guide to Building Dependable Distributed Systems, 2nd edn. Wiley, Chichester (2008)
2. Balser, M., Bäumler, S., Knapp, A., Reif, W., Thums, A.: Interactive Verification of UML State Machines. In: Davies, J., Schulte, W., Barnett, M. (eds.) ICFEM 2004. LNCS, vol. 3308, pp. 434–448. Springer, Heidelberg (2004)
3. Busch, M.: Integration of Security Aspects in Web Engineering. Master's thesis, Ludwig-Maximilians-Universität München (2011), http://uwe.pst.ifi.lmu.de/publications/BuschDA.pdf
4. Busch, M., Koch, N.: MagicUWE – A CASE Tool Plugin for Modeling Web Applications. In: Gaedke, M., Grossniklaus, M., Díaz, O. (eds.) ICWE 2009. LNCS, vol. 5648, pp. 505–508. Springer, Heidelberg (2009)

5. Clavel, M., da Silva, V., Braga, C., Egea, M.: Model-Driven Security in Practice: An Industrial Experience. In: Schieferdecker, I., Hartman, A. (eds.) ECMDA-FA 2008. LNCS, vol. 5095, pp. 326–337. Springer, Heidelberg (2008)
6. Gilmore, S., Gönczy, L., Koch, N., Mayer, P., Tribastone, M., Varró, D.: Non-functional Properties in the Model-Driven Development of Service-Oriented Systems. J. Softw. Syst. Model. 10(3), 287–311 (2011)
7. Gnesi, S., Mazzanti, F.: On-The-Fly Model Checking of Communicating UML State Machines. In: Proc. 2nd ACIS Int. Conf. Software Engineering Research, Management and Applications (SERA 2004), Los Angeles (2004)
8. Hafner, M., Breu, R.: Security Engineering for Service-Oriented Architectures. Springer, Heidelberg (2008)
9. Holzmann, G.J.: The SPIN Model Checker: Primer and Reference Manual. Addison–Wesley, London (2004)
10. Jürjens, J.: Secure Systems Development with UML. Springer, Heidelberg (2004); Tools and further information, `http://www.umlsec.de/`
11. Knapp, A., Merz, S., Rauh, C.: Model Checking - Timed UML State Machines and Collaborations. In: Damm, W., Olderog, E.-R. (eds.) FTRTFT 2002. LNCS, vol. 2469, pp. 395–416. Springer, Heidelberg (2002)
12. Koch, N., Knapp, A., Zhang, G., Baumeister, H.: UML-based Web Engineering: An Approach based on Standards. In: Web Engineering: Modelling and Implementing Web Applications. Human-Computer Interaction Series, pp. 157–191. Springer, Heidelberg (2008)
13. Lodderstedt, T., Basin, D., Doser, J.: SecureUML: A UML-Based Modeling Language for Model-Driven Security. In: Jézéquel, J.-M., Hussmann, H., Cook, S. (eds.) UML 2002. LNCS, vol. 2460, pp. 426–441. Springer, Heidelberg (2002)
14. Meliá, S., Gómez, J., Pérez, S., Díaz, O.: A Model-Driven Development for GWT-Based Rich Internet Applications with OOH4RIA. In: Proc. 8th Int. Conf. Web Engineering (ICWE 2008), pp. 13–23. IEEE, Los Alamitos (2008)
15. Menzel, M., Meinel, C.: A Security Meta-model for Service-Oriented Architectures. In: Proc. 2009 IEEE Int. Conf. Services Computing (SCC 2009), pp. 251–259. IEEE, Los Alamitos (2009)
16. Moreno, N., Fraternali, P., Vallecillo, A.: WebML modelling in UML. IET Software 1(3), 67 (2007)
17. Schumacher, M.: Security Engineering with Patterns: Origins, Theoretical Models, and New Applications. LNCS, vol. 2754. Springer, Heidelberg (2003)
18. Valverde, F., Pastor, O.: Applying Interaction Patterns: Towards a Model-Driven Approach for Rich Internet Applications Development. In: Proc. 7th Int. Wsh. Web-Oriented Software Technologies, IWWOST 2008 (2008)
19. Zhang, G., Hölzl, M.: Aspect-Oriented Modeling of Web Applications with HiLA. In: Wsh. Proc. 11th Int. Conf. Web Engineering (ICWE 2011). LNCS. Springer, Heidelberg (to appear, 2011)

Quality of Health Web Sites: Dimensions for a Wide Evaluation

Álvaro Rocha[1], Avelino Victor[2], and Patrícia Leite Brandão[3]

[1] Universidade Fernando Pessoa, Praça 9 de Abril, 349, 4249-004 Porto, Portugal
amrocha@ufp.edu.pt
[2] Instituto Superior da Maia, Av. Carlos Oliveira Campos - Castelo da Maia
4475-690 Avioso S. Pedro, Maia, Portugal
javemor@ismai.pt
[3] Instituto Politécnico do Cávado e do Ave, Campus do IPCA - Lugar do Aldão,
4750-810 Vila Frescainha S. Martinho, Barcelos, Portugal
plbrandao@ipca.pt

Abstract. The use of a methodology for the evaluation, comparison and quality improvement of Health Web Sites is justified by its widespread adoption and visibility to Internet users. Due to the sensitiveness of their content and impact on users, health related sites should be evaluated. This chapter proposes three different dimensions for the development of quality evaluation methodologies of Health Web Sites: contents, services and technical. We consider that these dimensions should be addressed transversally, providing an integrated and better overall evaluation.

Keywords: Health Web Sites, Quality Dimensions, Quality Evaluation.

1 Introduction

As citizens we are integrated in a global market that includes Web services and contents. We are currently in a "Global Era", the era of Knowledge and Information Society. Most of us have a fast access to these services and contents, mostly at the distance of a click.

Beyond the availability of contents, the technology development allows people to make medical appointments, exams and treatments or simply look for support and advices from health care professionals online, at any time of the day, any day and anywhere. This reality allows health care institutions to provide new and innovative services to their patients.

The exponential growth of the Internet gave place to a widespread search for contents and online services, including those concerning health issues. This search is reflected on the fact that most Internet users access the Internet for health issues questions [3, 7, 8, 9], demanding a vast number of related Web Sites [1].

Although Web Sites Quality is a pertinent question in every business area, the sensitiveness, sensibility and importance associated with the health care area justify an attentive observation of this subject [4].

J. Grabis and M. Kirikova (Eds.): BIR 2011, LNBIP 90, pp. 254–266, 2011.

In the present chapter, the Health Care Web Sites and their importance are approached, as a starting point to the development of a wide methodology. The evaluation, comparison and improvement of the Health care institutions' Web Sites is transversal to the main quality dimensions: Contents, Services and Technical.

Thus, in the following sections we describe Health care Web Sites, identifying their different categories. We present statistics concerning their existence and search, discussing their importance and need for satisfying quality requisites. Finally, we propose a high-level structure for a global quality evaluation of a Web Site.

2 Health Web Sites

2.1 Web Site Definition

According to the World Wide Web Consortium (W3C) a Web Site is a "set of connected Web pages, including a main page, located in the same net. Pages are accessed following a sequence, starting on a main page and ending on the wanted page." [2].

We can then define Web Site as electronic pages that contain connections, images and objects, with a code interpreted by a browser. Meaning, it is a set of contents and services remotely accessible through an HTTP communication protocol, composed by HTML static and/or dynamic pages and other multimedia elements. This is what makes a World Wide Web (WWW).

2.2 Health Web Site Definition

A Heath Web Site is an Internet location that allows the access to contents and services related to Health. Examples of this are education and health prevention, as well as booking management for health services and health acts payments.

One of the main questions from health and healthcare websites concerns their services and contents quality guarantee [4].

2.3 Health Web Sites Categories

Health Web Sites have distinctive focus, domain, destinations, objectives and functionalities. This fact suggests the search for the establishment of Health Web Sites' categories, but literature is poor in this subject. The literature presents the two founded classifications.

In the first classification, Giacomo e Maceratini [5] frame Health Web Sites in four categories:

- Health Portal – Web Site with online services and/or pharmacies.
- Documental Site and Meta-Site – Web Site with connections and/or references to other sites.
- Health Professionals and Citizens Site – mainly concerning specific diseases. Examples can be found in Medscape and Medconsul.
- Doctors' personal Sites – Medical Doctors' Web Sites including information concerning addresses and schedules, specialization, appointments proceedings, email, etc.

The second classification provides a wider categorization from the Health Improvement Institute [6]. It classifies Health Web Sites in two detail levels, being the higher level composed by seven categories:

- Communication Web Site – in order to communicate health contents to consumers, health professionals, or other specific individual groups, including:
    - Health Web Sites Search;
    - Health information/research Web Sites resources (books, papers, etc.);
    - Web Sites with health information (education, prevention, drugs descriptions, etc.);
    - Online health advices (FAQs, online chats, etc.);
    - Web Sites with Health Web Sites' rankings/evaluations;
    - Decision Support Web Sites.
- Behavioral Modification Web Sites – to support individuals that intend to change or control their behavior. For example, stop smoking, lose weight or exercise regularly. Including:
- Personal support Web Sites;
    - Disease management Web Sites.
- Online products Web Sites – to communicate information on health products and/or sell such products to consumers and/or health professionals.
    - Online Pharmacies;
    - Online Chemists;
    - Health products' marketing Web Sites.
- Health care institutions' Web Sites.
    - Health plans institutions' Web Sites;
    - Health care providers' Web Sites (hospitals, health centers, clinics, etc.);
    - Health auxiliary Institutions' Web Sites (Labs, blood and tissue banks, ambulances, etc.);
    - Health resources providers' Web Sites (equipment, personnel, etc.);
    - Health associations' Web Sites (associations, societies, etc.).
- Public health Web Sites (programs, legislation, etc.).
- Research Web Sites (recruiting people to develop health/clinical research).
- Other Web Sites Categories.

In practice, Web Sites can have characteristics from two or more second level categories, although the first level category is the one that classifies a Web Site.

During the research, we will adopt this Health Improvement Institute [6] classification and develop an evaluation, comparison and improvement methodology of health care institutions, specifically focusing on health care providers.

3 Health Web Sites Statistics

With the intent of evaluating the Internet Health pages availability, we will base this work on the current three main search engines. Thus, we searched on Google, Yahoo and Bing for Health and a set of terms that normally represent the countries' governmental areas. As a comparison, we will also search other terms from different subjects: Tourism, Economy, Finances, Education, Law, Industry, Agriculture and Defense.

Table 1 shows the results for the quantity of indexed pages for each term, in a descending way, according to the number of results provided by Google search engine.

The term "Health" is, among the searched ones, returned the highest number of indexed pages in the three used search engines, due to the existence of a high number of health Web Sites on the Internet.

According to the report "Internet Access and use in the EU27 in 2008" [7], 28% of European Internet users accessed the Web to search for information on health, behind Internet-Banking (29%) and travel and accommodation (32%). If we focus on Portugal only, the Web Access to search for health information comes first with 22% of the total searches, followed by online newspapers and magazine editions searches (20%), contacts with public administration (18%) and Internet-Banking (14%).

Table 1. Quantity of indexed pages on January 7th, 2010

Term	Google	Yahoo	Bing
Health	929.000.000	6,010,000,000	612,000,000
Education	684.000.000	5,210,000,000	513,000,000
Industry	505.000.000	2,590,000,000	351,000,000
Law	480.000.000	2,270,000,000	319,000,000
Economy	217.000.000	1,310,000,000	105,000,000
Defense	154.000.000	890,000,000	79,000,000
Agriculture	139.000.000	701,000,000	54,600,000
Tourism	130.000.000	630,000,000	80,500,000
Finances	56.000.000	307,000,000	204,000,000

According to the Health Trends Across Europe 2005-2007 study [8], the users of Health Web Sites raised from 44% in 2005 to 54% in 2007, on the average of seven European countries. The growth on the Internet use to access Health Web Sites was evident in all the participant countries of this study. Although, as seen on Table 2, the differences between Northern Europe (Norway, Denmark and Germany), Southern Europe (Greece and Portugal) and the Eastern Europe (Poland and Latvia) countries is significant.

In 2009, the study "Cybercitizen Health® Europe v9.0" [3] indicates that 85% of five European countries' Internet users (United Kingdom, Germany, France, Italy and Spain) used Internet to access health information.

Table 2. Internet users and health web sites users – adapted [8]

Country	% Internet Users		% Health Area Web Sites Users	
	2005	2007	2005	2007
Denmark	81	87	62	72
Norway	80	88	59	67
Germany	69	65	49	57
Latvia	53	67	35	47
Poland	53	67	42	53
Greece	42	47	23	32
Portugal	49	52	30	38
Average	71	83	44	54

Focusing on non-European countries, the results of the "Online Health Search 2006" from Pew Foundation [9], indicates that 75-80% of the USA Internet users searched for health related information.

Yellowlees' (2008) [1] estimates for the entire world the existence of more than 200,000 Web Sites entirely dedicated to health and that 60% of Internet users accessed it for health related issues.

Summarizing, the presented indicators take us to the conclusion that the Health term is founded on a variety of Web pages indexed by Google, Yahoo and Bing and that the other hand health Web Sites are the mostly searched by Internet users from Portugal and other countries, including European and American ones.

4 Health Web Sites' Importance

There are several potential benefits from using the Internet. To health professionals, Internet can be another mean through which they can exchange contents with other professionals or patients, as well as another source of information with continuous growth [10]. Internet can also facilitate the contact between the health care provider and the emotional support to patients, their relatives and/or friends.

The growing access to Web-based contents raises concerns regarding their quality and impact they have in users. [11]. A study from RAND to the California Healthcare Foundation [12], showed that the existing contents on Health Web Sites are frequently incomplete or obsolete.

The contents around health may be displayed on the Web by anyone with Internet Access and interest in the subject. Many of those sources are valid and credible, others may be well-intended but wrongly informed, others can be malicious and misleading.

Hence, the health contents' quality on the Internet is extremely variable and difficult to evaluate.

Users can access the Internet to obtain information on health and medical conditions, and the availability of such content allows visitors to assume much more responsibility towards their own health. At the same time it raises concerns that need to be assessed. The contents on health available to Internet users can be inaccurate or outdated. Furthermore, there are Web Sites offering biased contents that could be developed by a person or organization with commercial purposes. Another risk is that visitors can consult contents outside the context or misapplied to their own health conditions. Acting based on those contents without checking first their validity with a qualified health professional may have harmful consequences to the user [4].

The Internet advantages as an information source on health includes convenient access to a vast volume of contents, facility in contents' update, possibility of interactive formats that allow information understanding and storage. The Internet contents on health can make patients more informed, leading to a better health service, a more adequate use of health services related resources, and a stronger doctor-patient relation. Health professionals, however, inform about the discredit of the online available contents due to the absence of an effective mechanism to erase questionable cures, representing a risk to human health.

Thus, although there is a wide variety of online contents and services for health, often there can be no guarantees regarding their quality, as proved previously [4].

Some diseases that are still relatively rare and cannot be easily diagnosed, treated and understood. This way, the visitors' ability to find reliable contents becomes a bigger challenge. Some visitors truly believe on Internet available contents.

For all these reasons, the availability of quality health contents becomes a central question in the health informatics domain. It is extremely important that Health Institutions pay special attention to the development of their own Web Sites, considering the necessary quality of contents, services or technical requisites.

An example of this availability of quality contents is the fact that a group of medical doctors from the New Southeastern University found several mistakes in Wikipedia texts related to more than 80 drugs. The records found did not include important information, such as, for example, the fact that the anti-inflammatory Arthrotec causes abortion on pregnant women or that St John's Herb can interfere with the Prezist effect, a drug used to treat AIDS [13].

Kevin Clauson, a medical doctor from the University of Florida alerts to the fact that people who base «only on those data and do not consult a health Professional take a big risk of ingesting drugs inappropriately». «Those gaps can be as dangerous as a medical prescription mistake», said the specialist. According to the medical doctor, the pharmaceutical lab representatives have been erasing Wikipedia information, making it look unsafe. After 90 days of analysis specialists become aware that the Wikipedia texts on drugs received a «marketing bath» [13].

Another example is the World Health Organization (WHO) alerting to the fact that one in each ten drugs commercialized on the Internet is false, making the percentage raise up to 50% in poor countries. By 2010, the business volume of false pharmaceutical drugs will be, according to Bate's projections [14], of 51,5 million Euros, up to the world scale, representing an increase of 90% since 2005. According to Bate [14], "either it is a false Viagra or false cancer drugs, thousands of people die everyday due to China and India counterfeiters that mix chalk, powder and dirty water to make drugs and sell them all around the world. With Internet becoming a global storeroom, these poisoning drugs can come to a pharmacy next to you". During the last ten years, the false drugs traffic became the faster growing business in the world, adds the author.

According to Peter Yellowlees [1], appointments with medical doctors will be done through Internet, turning this service into a "common place". On the safety area and due to their own convenience, patients will be able to interact with health professionals through their own homes.

Internet poses a risk to health information seekers, through unreliable information or wrong advice. Online false contents and services must be identified in order to avoid damages. [1].

Potential risks that patients using the Web as a source of information are, according to Peter Yellowlees [1]:

- Virtual world excessive dependence;
- Bad advices, especially from anonymous Internet users;
- Bad quality of contents and services;
- Financial prejudice, through wrong therapeutics, or therapies, and untested health products;
- Dishonest relationship.

Our intention for this research is to develop an adequate methodology to users, owners and developers to being able to evaluate, compare and improve quality on Health Web Sites belonging to health care providers, with a criteria that can be considered or classified inside a specific necessity hierarchy.

5 Software Quality

People look for quality in each object they create, and software is not an exception. Software is one of the strategic assets in the Information Society. With the Internet boom, and the following exponential increase in contents and services made available through websites, a quality revolution quickly spread throughout the whole world [34].

Aspects related to the quality of websites have, therefore, become relevant to many sectors of activity. Several contributions to the field of websites quality and different schools of thought have primarily focused in the definition of quality, in its structure and in the way it can be measured [e.g., 23, 32].

In this chapter we adopt the definition of quality published in the most recent ISO (International Organization for Standardization) standard for software quality, because it agrees with our purposes, because of its broadness and completeness and because of the prestige of the mentioned organization. We, therefore, understand quality as the "*capability of a software product to satisfy stated and implied needs when used under specified conditions*" [28].

6 Dimensions of Websites Quality

The content and services are the reasons for the existence of a Web site. Thus, considering the results of some studies conducted and/or oriented by one of the authors

[41], as well as the systematization of the knowledge available in several bibliographies, we can group websites quality in three main dimensions (Fig. 1): Content quality; Service quality; and Technical quality.

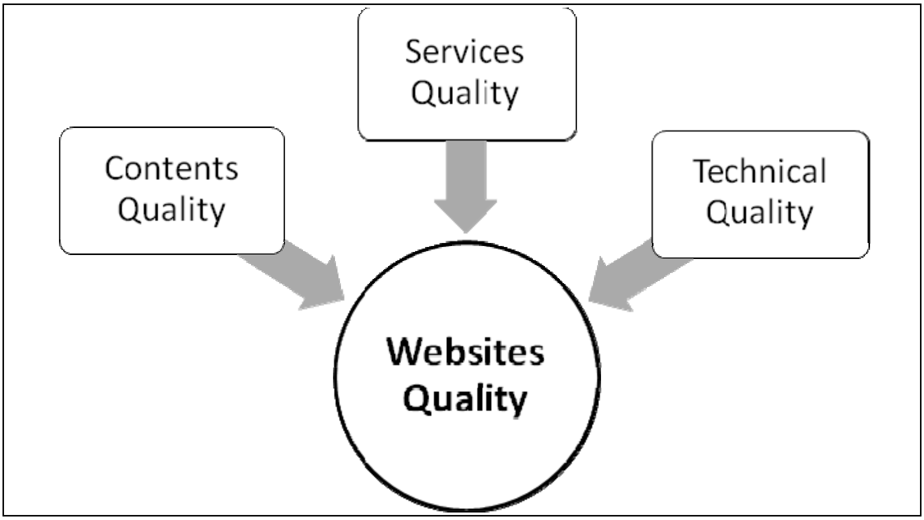


Fig. 1. Main Dimensions for Websites Quality

In the first dimension, the main concerns focus in contents quality and not in its existence, this should be a technical quality concern. In contents quality, attributes like accuracy/precision, completeness, relevance, opportunity, consistence, coherence, update, orthography and syntax are evaluated.

In the second dimension, concerns are focused in the quality of services offered in websites. In services quality, attributes like security, reliability, privacy, performance, efficiency, accuracy, opportunity, availability, response time, time saving, empathy, reputation and personalization are evaluated.

Finally, in the third dimension, concerns are focused on the technical quality of websites, i.e., on quality attributes that are usually in quality standards for software, such as ISO/IEC 9126 [26] and its successor ISO/IEC 25010 [24]. Thus, attributes like navigation map, path, search engine, download time of pages, browser compatibility, broken links and accessibility are evaluated.

Through bibliography and through our experience we have observed that there is no evaluation methodology that focuses in these three main websites quality dimensions, in a broad, integrated and transversal sense.

6.1 Contents Quality Evaluation

Content quality evaluation primarily employs methodologies based in the Likert scale, which evaluates the quality of these contents amongst respondents (users, linguists and experts in the contents presented in websites). Some studies related to this dimension are worth mentioning, such as: Bernstam et al. [17], Caro et al. [18] Hargrave et al. [21], Moraga et al. [33], Parker et al. [37] and Richard et al. [40].

A possible structure for a contents quality measuring instrument can have a similar format to the one presented by Moraga et al. [33], already consistent with the ISO 25012 standard [25], which is the most relevant contribution that we found in the literature for this dimension. This structure is composed of points of view (inherent, system dependent), categories (intrinsic, operational, contextual, representational), characteristics and sub-characteristics.

To classify each attribute, the analyzers should carefully read and analyze either every webpage content of the website or every webpage content until a certain predefined level of depth is reached.

Analyzers can classify webpage content quality characteristics and sub-characteristics in a five point Likert scale (1- Bad, 2 – Mediocre, 3 – Reasonable, 4 – Good, 5 – Very Good).

6.2 Services Quality Evaluation

Online services quality evaluation, which includes, for instance, in the health area, healthcare acts scheduling, prescriptions renewal or drug acquisition, usually employs evaluation methodologies for back-office procedures and/or users' satisfaction towards services available in websites. Some studies related to this dimension are worth mentioning, such as: Al-Momani and Noor [15], Arshad et al. [16], Cernea et al. [19], Hamadi [20], Li and Suomi [29], Li and Suomi [30], Parasuraman et al. [38] and Zhao [42].

A possible structure for a services quality measurement instrument can have the format based in Parasuraman's services quality scale [38], which is the most relevant contribution that we found in the literature for this dimension. This structure is composed of characteristics (efficiency, system availability, fulfillment, privacy, responsiveness, compensation, contact) and attributes.

Analyzers can classify websites services quality attributes in a 5 point Likert scale (1 – Completely disagree; 2 – Disagree; 3 – Don't agree or disagree; 4 – Agree; 5 – Completely agree).

6.3 Technical Quality Evaluation

The technical dimension is related to how the content and services are assembled and made available on a Web site.

Technical quality evaluation is based in software quality models or standards and in methods focused in usability, methods developed through studies in the area of Human-Computer Interaction (HCI). The first group methodologies include, amongst others, ISO/IEC 9126 [26] and ISO/IEC 25010 [24] standards. The second group includes an approach that emerged with the hypermedia nature of the Internet and the relevance of interface conception to speed access to information and globally improve human-computer interaction, and includes standards like ISO/IEC 9241 [27]. This approach defines quality according to usability, considering the users' point of view [e.g., 22, 35].

Technical quality is, amongst the three main dimensions, the one that has received a higher degree of attention from researchers, and several methodologies have been proposed for its evaluation. Amongst those, we consider the work developed by Olsina [36] and other works that followed this line [e.g.: 31, 39] the most relevant, since they base their methodologies in ISO 9126 [26] and its high level quality characteristics that interest websites users. This structure is composed of characteristics (usability, functionality, reliability, efficiency), sub-characteristics and attributes.

Technical quality measurement can also be classified in a three or five point Likert scale for each quality attribute.

7 Structure for a Global Quality Evaluation of a Health Website

Bearing in mind the three main dimensions for websites quality defined in this chapter, resulting of literature systematization and based in our experience as websites users and websites engineering researchers, we now propose a high-level structure to evaluate global quality of a health website in a broad, transversal and detailed form.

This structure is organized according to the three main website quality dimensions, comprised by characteristics which, in their turn, are comprised by attributes, as shown in Fig. 2. Characteristics can sometimes be comprised by more than one level of sub-characteristics.

Websites quality evaluation, comparison and improvement methodologies developed through the proposed structure should be designed to incorporate adjustments to the activity sector in which they are applied, since the suitable structure for quality sub-characteristics and attributes generally differs between activity sectors. Simultaneously, they must be configured without the existence of overlap between the characteristics, sub-characteristics and attributes of the three dimensions.

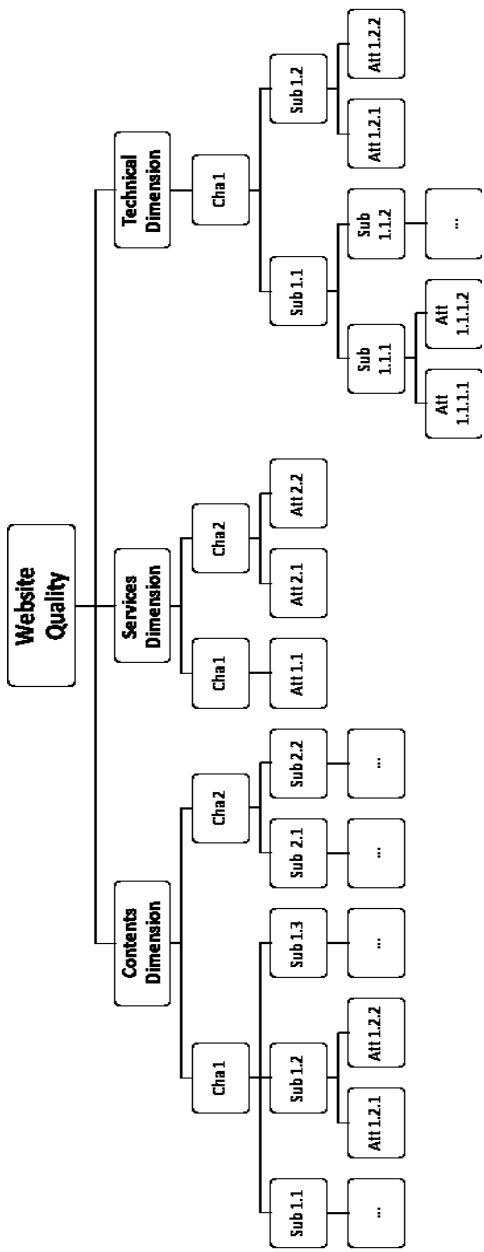
8 Final Remarks

In this chapter we approached general Web Sites, with particular focus on Health Web Sites. For these last ones, we analyzed some statistic indicators of availability, search and importance.

We have confirmed that the complexity and purpose of Web Sites are diverse, and that Health Web Sites are the most searched ones in Portugal and in other countries of reference. Furthermore, these exist in a large number and have critical importance, due to their negative implication on citizens' health when the quality of information and services provided is low or unacceptable.

If Web Sites' quality is an important issue in any business area, the sensitivity and importance given to health-related issues justify a more attentive and thorough observation of the related areas.

In this chapter we proposed, also, a high-level structure for a global quality evaluation of a health website. We can highlight a few aspects as a conclusion:



Legend: Cha - Characteristic; Sub - Sub-characteristic; Att - Attribute

Fig. 2. Part of the high-level structure for a global quality evaluation of a health website

1. Websites quality is strategically important for organizations and for the satisfaction of their clients;
2. Websites quality should be based in the quality measurement the three main dimensions: content, services, and technical;
3. A structure based in these three dimensions, characteristics, sub-characteristics and attributes will substantiate a broad, integrated, transversal and detailed quality evaluation of a website;
4. A good evaluation, comparison and improvement methodology for websites quality should properly comprise the three mentioned quality dimensions and allow the adjustment to a specific activity sector.

The next step in our study will be the development of an evaluation, comparison and improvement methodology for the quality of institutional and hospital websites, based in the high-level structure proposed in this chapter and will be built and validated with help of Web engineering experts and hospital websites users. The need for this methodology is justified by the fact that we do not know any that provides a broad and detailed assessment, integrating the three main quality dimensions of a website: content, services and technical.

References

1. Yellowlees, P.: Your Health in the Information Age - How you and your Doctor Can Use The Internet To Work Together, iUniverse, USA (2008)
2. Lavoie, B., Nielsen, H.F.: Web Characterization Terminology & Definitions Sheet. In: W3C (1999)
3. Manhattan Research: Cybercitizen Health® Europe v9.0, eHealthNews.eu., `http://www.ehealthnews.eu/content/view/1823/26/` (accessed December 16, 2009)
4. Lorence, D., Abraham, J.: A study of undue pain and surfing: using hierarchical criteria to assess website quality. Health Informatics Journal 14(3), 155–173 (2008)
5. Giacomo, P.D., Maceratini, R.: Health websites in Italy: use, classification and international policy. Methods of Information in Medicine 27, 153–160 (2002)
6. H.I. Institute: Health Website Ratings Instrument (HWRI): Taxonomy of health websites, Health Improvement Institute (2006)
7. Eurostat: Internet Access and use in the EU27 in 2008. Statistical Office of the European Communities (2008)
8. Andreassen, H.K., Sørensen, T., Kummervold, P.E.: eHealth Trends across Europe 2005-2007, Norwegian Centre for Telemedicine (2007)
9. Fox, S.: Online Health Search 2006, PEW / Internet and American LIife Project, Washington (2006)
10. Larcher, L.M., Juzzo, C.: Critérios para avaliação da qualidade das informações sobre saúde disponiveis online. In: Anais do XVIII Congresso Brasileiro de Engenharia Biomédica (2005)
11. Karp, S., Monroe, A.: Quality of healthcare information on the Internet: caveat emptor still rules. Managed Care Quarterly 10(2), 3–8 (2002)
12. CHCF: Proceed with caution: a report on the quality of health information on the Internet, California HealtCare Foundation (2001)
13. Clauson, K.A., Polen, H.H., Boulos, M.N., Dzenowagis, J.H.: Scope, Completeness, and Accuracy of Drug Information in Wikipedia. The Annals Pharmacotherapy 42(12), 1814–1821 (2008)

14. Bate, R.: O Mundo Fatal dos Medicamentos Falsificados. In: FP - Foreign Policy, Edição FP Portugal (2008)
15. Al-Momani, K., Noor, N.: E- Service Quality, Ease of Use, Usability and Enjoyment as Antecedents of E-CRM Performance: An Empirical Investigation in Jordan Mobile Phone Services. The Asian Journal of Technology Management 2(2), 11–25 (2009)
16. Arshad, N., Ahmad, F., Janom, N.: Empirical Study on Electronic Service Quality. Public Sector ICT Management Review 2(1), 29–37 (2007)
17. Bernstam, E.V., Shelton, D., Walji, M., Meric-Bernstam, F.: Instruments to assess the quality of health information on the World Wide Web: What can our patients actually use? International Journal of Medical Informatics 74, 13–19 (2005)
18. Caro, A., Calero, C., Caballero, I., Piattini, M.: A proposal for a set of attributes relevant for Web portal data quality. Software Quality Journal 16, 513–542 (2008)
19. Cernea, S.O., Sîrbu, J., Mărginean, N.: Determination of Users Satisfaction Level Regarding the Quality of E-Services Provided by Bogdan-Voda University. Annales Universitatis Apulensis Series Oeconomica 11(2), 652–661 (2009)
20. Hamadi, C.: The Impact of Quality of Online Banking on Customer Commitment. Communications of the IBIMA, 1–8 (2010)
21. Hargrave, D.R., Hargrave, U.A., Bouffet, E.: Quality of health information on the Internet. Neuro-Oncology 8, 175–182 (2006)
22. Hinchliffe, A., Mummery, W.K.: Applying usability testing techniques to improve a health promotion website. Health Promotion Journal of Australia 19(1), 29–35 (2008)
23. Ho-Won, J., Seung-Gweon, K., Chang-Shin, C.: Measuring Software Product Quality: A Survey of ISO/IEC 9126. IEEE Software 21(5), 88–92 (2004)
24. ISO. ISO/IEC 25010:2011: Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – System and software quality models. International Organization for Standardization (2011)
25. ISO/IEC. ISO/IEC 25012:2008: Software engineering – Software product Quality Requirements and Evaluation (SQuaRE) – Data quality model, International Organization for Standardization (2008)
26. ISO/IEC. ISO 9126-1:2001: Software engineering – Product quality – Part 1: Quality model, International Organization for Standardization (2001)
27. ISO/IEC, ISO 9241-11:1998: Ergonomic requirements for office work with visual display terminals (VDTs) – Part 11: Guidance on usability, International Organization for Standardization (1998)
28. ISO/IEC: ISO/IEC 25000: Software Engineering – Software product Quality Requirements and Evaluation (SQuaRE) – Guide to SQuaRE. International Organization for Standardization (2005)
29. Li, H., Suomi, R.: Evaluating Electronic Service Quality: A Transaction Process Based Evaluation Model. In: Proceedings of ECIME 2007: The European Conference on Information Management and Evaluation, Montpellier, France, September 20-21, pp. 331–340 (2007)
30. Li, H., Suomi, R.: A Proposed Scale for Measuring E-service Quality. International Journal of u- and e-Service, Science and Technology 2(1), 1–10 (2009)
31. Machado, R., Rocha, A.: Avaliação da Qualidade de Sítios Web Institucionais: Aplicação de Métrica às Faculdades de Medicina do Espaço Ibérico. Revista da Faculdade de Ciência e Tecnologia da Universidade Fernando Pessoa 3, 76–87 (2008)
32. Mich, L., Franch, M., Gaio, L.: Evaluating and Designing Web Site Quality. IEEE Multimedia 10(1), 34–43 (2003)

Analysis and Evaluation of Selected Shops with Organic Food in Poland

Witold Chmielarz

Faculty of Management, University of Warsaw, Szturmowa 1/3 Str.,
02-678 Warsaw, Poland
witek@mail.wz.uw.edu.pl

Abstract. The main objective of this article is to analyse and evaluate the phenomenon of selling organic food via the Internet. In the beginning of this article the author described a specific character of the examined sector. Subsequently, he presented the basic assumptions of the study. Next, the author carried out research and performed an analysis of findings by means of a scoring method and a scoring method with a preference scale.

Keywords: electronic commerce, websites, organic food.

1 Introduction

At the turn of 2010 and 2011 the author performed a comprehensive analysis of internet services in the food sector in Poland. The study included different types of internet grocery shops as well as specialized food sector vortals and websites of food producers [1]. This study is a specific supplement of the earlier research conducted by the author. The present study concerns sales of organic food to individual clients, the area which is becoming increasingly popular among consumers and developing at a very fast pace.

There are many publications concerning the problem of websites' evaluation, especially in financial sector [11, 12, 13]. The review of the literature shows that e-commerce websites may be analysed from the point of view of: usability (site map, directory); functionality (search, navigation, relevance of content); visualisation (colour scheme, background, graphics, text); reliability and availability.

General assumptions of most methods of websites' evaluation are usually based on the models taken from electronic commerce [3, 14, 16]. Most of them are traditional scoring methods based on specific criteria sets, evaluated with an applied scale. The most commonly applied criteria are technical and functional ones. Most of them contain factors which may be evaluated in a very subjective way: text clarity, attractive colours, images and pictures. There occur frequent problems with determining preferences from the point of view of a final user (customer) or expert who is evaluating this phenomenon [2]. This article deals with these issues and it will be useful for researchers and practitioners who look for solutions to these problems. But this work mainly concerns the application of the author's own, though based on the literature, set of criteria for a scoring evaluation and a selection of electronic services of selected shops with organic food in Poland.

J. Grabis and M. Kirikova (Eds.): BIR 2011, LNBIP 90, pp. 267–280, 2011.

By definition, *organic farming* is described as *…a specific form of management and food production, where food is produced with natural methods in clean and safe environment, without artificial fertilizers and synthetic plant protection products, antibiotics, growth hormones and genetically modified organisms…* [10]. And the term *organic food* is attributed to food produced by organic farming methods [7]. The number of consumers of organic food is growing from year to year so, proportionally, both the demand for organic produce and its production increase. In 2009 about 35 million ha of land in 154 countries were under organic management, and the production of organic food exceeded the value of 50 billion dollars. According to official statistics [5] annual increase of the area managed in accordance with organic methods in the world is 2% [9]. Only once, in 2005-2006, we observed a fall of 1%. The statistics show that when we take into account the total utilised agricultural area used for organic production in 2009 the leaders were: Oceania (32.6%), Europe (24.9%) and Latin America (23%). The countries with the largest area of organic crops at the time were: Australia (12%), Argentina (4.4%) and the USA (1.95%), and among European countries: Spain (1.33%). However, the highest share of organic farming was in the Falklands (35.7%), Liechtenstein (26.9%) and Austria (18.5%). Among former socialist countries, in Estonia 10.5% of the total utilized agricultural area is used for organic production, and in Czech Republic it amounts to 9.4%. The highest number of organic food producers is in India (over 600,000), Uganda (187,000) and Mexico (128,000), and among the European countries organic farming is widely popular in Spain (25,000). However, the distribution of organic produce markets is different: the biggest market is in the USA, it amounts to nearly 18 billion euros (increase from 2008 to 2009, 5.1%), that is 3.7% of the total sales of food and beverages, and the proportion of organic fruit and vegetables constitutes 11.4% of the total sales of fruit and vegetables in the USA. Subsequently, the market is dominated basically by European countries: Germany, France, Great Britain, and they are followed by Spain and Austria. In the top ten, among the non-European countries there are only two states: Canada and Japan. However, the greatest consumption per capita is observed mainly in Europe: organic produce is very popular in Denmark, Switzerland, Lichtenstein, Luxemburg, Sweden, Germany and France (i.e. these countries are regarded as the most industrialized, the most ecologically damaged and the richest). Among the non-European countries, the USA takes the eighth position, and Canada is in the tenth place.

Poland is lagging behind in the quoted statistics (from the 25th to 30th position, depending on the ranking); however, the data on organic food production look promising, to say the least. According to Agricultural and Food Quality Inspection Main Inspectorate (Główny Inspektorat Jakości Handlowej Artykułów Rolno-Spożywczych) at the end of 2007 the total number of organic farms in Poland amounted to almost 12 thousand, and since then we observed that the figure was on the increase. Over the next half a year it rose by over 3 thousand and by the end of 2008 by nearly 8 thousand [8]. Also, in shops there appears more and more food with Eco certificates. Recently, the largest range of organic products in Poland could be found in *ALMA, Piotr i Paweł, Jubilat, Stokrotka* or *Bomi* supermarkets. Among the biggest organic food producers we find: *Oleofarm* – vegetable oils, *Biopiekarnia* – bakery products, *Materne* – fruit processing, *O.K. Owocowe Koncentraty* – fruit processing, *Maurer* – freshly squeezed fruit, *Symbio* – producer of fruit, vegetables and cereal, *Bio Food* – fruit and vegetable processing and *Pol-Mak* – pasta manufacturing. It is estimated that organic food market in Poland will grow by 25-30% a year [4].

Organic food, despite a significant increase in its popularity in recent years, is still a niche product and it is not very accessible because of the limited number of outlets and high prices. We notice that the highest level of environmental awareness is among young people, regular users of the Internet and therefore, an online shop seems to be a perfect way to sell such products. It enables users to make purchases from every place in the country and it uses the medium which is popular among potential buyers. Additionally, it offers fast delivery of fresh produce at relatively low prices. However, online shops offering organic food are still a relatively new phenomenon and they are rarely encountered in Poland. The objective of the study is not only to analyse and evaluate the phenomenon but also to popularize the idea of selling organic food through the Internet, which, in turn, can lead to an increase in demand, and, potentially, also in the production of organic food in Poland whose environment is conducive to the dynamic development of the sector.

2 Assumptions of the Study

For the present evaluation the author has chosen seven shops specializing in selling organic food which appear as first among the search results of popular search engines and take leading positions in relevant rankings: www.biovert.pl; ekosfera24.pl; www.kozlek.pl; www.zdrowazywnosc.pl; www.pyszneizdrowe.pl; www.sklepekologiczny. pl and www.tobio.pl.

The author applied a scoring method to evaluate the shops. On the basis of earlier analyses of websites of grocery shops the author established criteria used for the evaluation of organic produce shops. The author performed a scoring evaluation of each criterion in a website according to the applied scale. The following scoring scale has been used: 0.00 – no feature, the highest costs; 0.25 – satisfactory criterion fulfilment; 0.50 – partial criterion fulfilment, medium costs; 0.75 – sufficient fulfilment of criterion and 1.00 – complete criterion fulfilment, the lowest costs.

Subsequently, for each website the author calculated the total of the points obtained in the assessment and analysed each specific case. In the next step he used a scoring method with preference points where weights (with the total of 1) have been assigned to particular criteria. The score is, in this case, the total of products of evaluations and weights:

It was concluded that, in contrast to the previously analysed grocery shops where we already find outlets with a well-established position, a sector of online shops with organic food may be seen as new and still developing. The criteria which have been applied (they were slightly modified in relation to those previously used for the evaluation of food sector) appear to properly reflect the specific character of the group. The established criteria used in the study were: technical (the ease of finding a product in the Internet (in search engines); availability of terms and conditions and delivery price lists, visualization (website design): (text readability – the size and colour of the font and its contrast against the background; graphics (background, contrast, font colour, colour matching); the quality of photographs; information display; menu layout; technological correctness; security of transactions); functional (the ease of finding a product in a website; the ease of navigation (the ease of product search, the availability of information); product range; methods of payments; selection of products; suggesting purchases of similar items (suggesting similar products – related content functionality)) and economic (price attractiveness; marketing factors (discounts, rebates)).

The evaluation consists of findings obtained from an experiment conducted with the fourth-year students (34 participants) of the Faculty of Management of the University of Warsaw during the classes on *Electronic Business* as well as expert estimates of the phenomena which are perceived as impossible to assess by a potential website user in the survey. Respondents participated in examining the following categories of the shops' evaluation: the ease of finding selected products, the ease of finding *Terms and Conditions* and *Delivery Terms*, visualization and product range.

In order to assess the first category, accessibility i.e. the ease of finding a shop in the Internet, the author examined the positions which the shops occupied in search results. For each shop the author has analysed the place it took in search results after entering the key words: *shop with organic food* into Google search engine, and, subsequently, the position was replaced with a scoring scale. After entering the name of a shop, each of the examined shops appeared as the first in the search engine, and the name appeared repeatedly among the search results (over 20 times it was displayed on the first page of the search engine).

Table 1. The position of the shop with organic food in Google search engine

Availability in the Internet	Position. Keyword: *Shop with organic food*	Score
biovert.pl	9	0.50
ekosfera24.pl	5	0.75
kozlek.pl	2	1.00
zdrowazywnosc.com.pl	25	0.25
pyszneizdrowe.pl	1	1.00
sklepekologiczny.pl	3	1.00
tobio.pl	6	0.75

Subsequently, the author analysed the ease of finding and purchasing the selected products according to the established measure. Similarly as in the situation presented in the thesis [15], the websites were evaluated on the basis of finding two selected products (grated/home-made beetroot, barley/rye flakes) and their search time. Then, the arithmetic mean of the time used by all people participating in the experiment was calculated and it was converted into a scoring scale. It turned out that the greatest problem occurred when researchers were trying to distinguish the products which would be found in the same form and made by the same producer in all examined shops.

Table 2. Evaluation of the speed of access to organic products in particular shops

The availability of selected products	Average time of finding a product. Product 1 in seconds	Average time of finding a product. Product 2 in seconds	Score – Product 1	Score - Product 2	Average score
biovert.pl	35.88	50.44	0.25	0.50	0.38
ekosfera24.pl	95.29	105.32	1.00	1.00	1.00
kozlek.pl	25.00	102.06	0.25	1.00	0.63
zdrowazywnosc.com.pl	61.32	0.00	0.75	0.00	0.38
pyszneizdrowe.pl	31.62	57.65	0.50	0.50	0.50
sklepekologiczny.pl	37.79	41.03	0.50	0.50	0.50
tobio.pl	25.88	28.68	0.25	0.25	0.25

Then, with the scale <0;1>, each person subjectively assessed the possibilities of finding terms and conditions concerning purchases, and they evaluated the availability of delivery terms. After calculating the average of these values, each time the results were converted into a scoring evaluation.

Similarly, the author performed the evaluation of visualization (website design) as well as the volume of the product range. The visualization was evaluated intuitively on the basis of the factors previously listed in the work: lettering, the colour of the background and elements, contrast, quality and layout of graphic elements and menu clarity.

The product range of the analysed shops was greatly diversified. Apart from organic products, many shops also offer non-organic food. Therefore, in the evaluation of the product range the author tried to analyse only organic produce. The evaluation of the product range has been performed by means of a survey because particular users may pay attention to various aspects of the range of available products.

The remaining aspects were evaluated by the author assisted by the experts cooperating with him. And, therefore, in the case of measuring user-friendliness and navigation, time needed to find particular products was the factor considered to be of greatest importance. All the remaining features were regarded as equally important.

The attractiveness of the prices of selected products was evaluated as at 3 January 2011 on the basis of the average price of the products of the Polish producers who are leaders in this category. These were the same producers as in the case of measuring the time needed for successful product search (grated/home-made beetroot, barley/rye flakes).

Table 3. Price attractiveness of selected products

Shop	Bio Food grated beetroot	Bio Food home-made beetroot	Bio Babalscy barley flakes	Bio Babalscy rye flakes	Average score of selected products	Score
biovert.pl	5.99		3.2	2.95	3.04	0.25
ekosfera24.pl	6.99	7.5	6.1	4.6	6.30	0.00
kozlek.pl	5.95			4.04	2.50	0.50
zdrowazywnosc.com.pl	4.99				1.25	1.00
pyszneizdrowe.pl		6.49	3.49	3.19	3.29	0.75
sklepekologiczny.pl	6.99	7.19	3.49	3.2	5.22	0.00
tobio.pl	4.99		2.54	2.49	2.51	0.50

The only comparable factors which can be classified as marketing activities in the examined shops consist in free delivery of purchases. A scoring scale was assigned respectively to discounts published in *Terms and Conditions*. Generally, we may observe that the promotions of goods change frequently and they are rather irregular: frequently the comment …*Brak produktów w promocji/No promotions*…is displayed (pyszneizdrowe.pl 21.01.2011).

Table 4. Marketing activities (discounts, rebates)

Shops	Discounts	Score
biovert.pl	Free delivery of purchases over 180 PLN and not exceeding 10 kg and transfer payments	0.75
ekosfera24.pl	Free delivery: Poczta Polska (Polish Postal Service) > 225 PLN and weight < 15kg; UPS - > 250 PLN and weight <24 kg; transport by Ekosfera > 120 PLN and weight <100 kg; personal collection > 20 PLN.	0.25
kozlek.pl	Free delivery of purchases over 250 PLN and within the territory of Poland and transfer payments	0.50
zdrowazywnosc.com.pl	Free delivery of purchases over 150 PLN and < 30 kg and transfer payment	1.00
pyszneizdrowe.pl	Free delivery: Poczta Polska (Polish Postal Service) > 250 PLN; courier within a specified area from >150 to > 350 PLN; courier – within the territory of Poland > 250 PLN	0.25
sklepekologiczny.pl	Free delivery: Poczta Polska (Polish Postal Service) > 400 PLN UPS> 500 PLN and transfer payment	0.25
tobio.pl	Free delivery of purchases > 250 PLN < 30 kg and transfer payment	0.75

Diversification of methods of payment attracts internet users. The methods of payment in selected shops with organic food are characterized below.

Table 5. Methods of payment in organic food shops

The name of the shop	Credit card payment	Transfer payment	Cash on delivery	Other	Score
biovert.pl	No	Yes	Yes	No	0.50
ekosfera.pl	No	Yes	Yes	Platnosci.pl, cash	0.75
kozlek.pl	No	Yes	Yes	Platnosci.pl, cash	0.75
zdrowazywnosc.com.pl	No	Yes	Yes	Cash	0.50
pyszneizdrowe.pl	Yes	Yes	Yes	Cash	0.75
sklepekologiczny.pl	Yes	Yes	Yes	No	0.75
tobio.pl	Yes	Yes	Yes	No	0.75

For clients the ease of accessing a particular product is also very important. In the case of online shops the ease of product search is guaranteed by efficient search applications. Some of the shops offer functionality where you may conduct a product search entering a product name and/or a description. In some cases you can also find a product using the name of the producer. The table below presents product search in examined online shops.

Table 6. Product search in organic food shops

Shop	Search by names and descriptions	Other	Score
biovert.pl	Yes	Selection of products by category and producer	0.75
ekosfera.pl	No/Yes	There is no possibility to search only in the product names, so in the search results there are products which contain the elements entered in the search application as their ingredients	0.25
kozlek.pl	Yes	No additional options	0.50
zdrowazywnosc.com.pl	Yes	Selection by category	0.75
pyszneizdrowe.pl	Yes	Selection by category and producer	0.75
sklepekologiczny.pl	Yes	Selection by category and producer	0.75
tobio.pl	Yes	Selection by category and producer. A possibility of selecting many categories in one search.	1.00

Another important aspect in using internet shops is related content functionality i.e. displaying similar products to the product or a group of products which is being viewed by a customer and/or the items which were purchased by the clients who also bought this particular product

Table 7. Related content (suggesting a product which is similar to the one being searched or ordered)

Shop	Related content	Score
biovert.pl	Other products purchased by the clients buying a particular product	0.75
ekosfera.pl	n/a	0.00
kozlek.pl	n/a	0.00
zdrowazywnosc.com.pl	n/a	0.00
pyszneizdrowe.pl	Other products purchased by the clients buying a particular product	0.75
sklepekologiczny.pl	Other products purchased by the clients buying a particular product	0.75
tobio.pl	Other products purchased by the clients buying a particular product or a similar product range	1.00

Another important aspect of using a shop is the security of transactions. We expect that after logging on, the connection is encrypted. However, such functionality is provided by one of the examined shops – ekosfera24.pl; in other shops we observe no such feature.

The last of the examined features of the website was its technological correctness. Despite certain controversies, it was examined with regard to correctness of HTML and CSS codes of the websites of online shops. The author has used the validators such as: http://validator.w3.org/ and http://jigsaw.w3.org/css-validator/. The results are shown in the table below.

Table 8. Technological correctness of examined websites

Technological correctness	HTML errors and comments	CSS errors and comments	Score
biovert.pl	65/15	3/25	0.50
ekosfera.pl	29/0	4/2	0.75
kozlek.pl	128/0	12/9	0.25
zdrowazywnosc.com.pl	211/5	176/15	0.00
pyszneizdrowe.pl	105/16	289/80	0.25
sklepekologiczny.pl	85/8	120/121	0.25
tobio.pl	10/3	12/201	1.00

3 Analysis of Collective Results

Averaged results of the survey and expert evaluations presented in the previous part of the article enables the author to carry out analyses based on the combined table.

Table 9. Evaluation of selected websites of food sector companies

Criteria/Internet shop	biovert.pl	ekosfera24.pl	kozlek.pl	zdrowazywnosc.com.pl	pyszneizdrowe.pl	sklepekologiczny.pl	tobio.pl	Total	%% maximum score
Availability in the Internet	0.50	0.75	1.00	0.25	1.00	1.00	0.75	5.25	75.00%
The ease of product search	0.38	1.00	0.63	0.38	0.50	0.50	0.25	3.63	51.79%
The ease of finding Terms and Conditions	0.54	0.92	0.88	0.70	0.76	0.79	0.93	5.53	78.99%
The ease of finding delivery terms	0.43	0.87	0.88	0.76	0.84	0.85	0.93	5.55	79.31%
Visualization	0.55	0.32	0.74	0.34	0.78	0.47	0.88	4.09	58.40%
Product range	0.88	0.77	0.78	0.37	0.57	0.63	0.91	4.90	70.06%
User-friendliness and navigation of a website	0.25	0.50	0.25	0.50	0.75	1.00	1.00	4.25	60.71%
Attractiveness of prices	0.25	0.00	0.50	1.00	0.75	0.00	0.50	3.00	42.86%
Marketing operations (delivery discounts)	0.75	0.25	0.50	1.00	0.25	0.25	0.75	3.75	53.57%
Methods of payment	0.50	0.75	0.75	0.50	0.75	0.75	0.75	4.75	67.86%
Search application	0.75	0.25	0.50	0.75	0.75	0.75	1.00	4.75	67.86%
Suggesting similar products (related content)	0.75	0.00	0.00	0.00	0.75	0.75	1.00	3.25	46.43%
Security	0.00	1.00	0.00	0.00	0.00	0.00	0.00	1.00	14.29%
Technological correctness	0.50	0.75	0.25	0.00	0.25	0.25	1.00	3.00	42.86%
Total	7.03	8.13	7.65	6.54	8.70	7.99	10.66	56.70	
%% maximum scores	50.21 %	58.09 %	54.6 2%	46.7 4%	62.13 %	57.04 %	76.16%		

The analyses carried out by the author point to the fact that generally organic food shops receive very low scores in the evaluation. Only two out of seven websites received scores above 60%. The highest number of points possible to obtain in the evaluation was assigned to tobio.pl website. The score which the shop received was 76.16% of the maximum score in the evaluation. This high rating of tobio.pl website is probably due to technological correctness, efficient search application, related content functionality (suggesting similar products), high scores for the product range as well as considerable ease of finding *Delivery Terms* and *Terms and Conditions* concerning purchases and deliveries. Unfortunately, the site does not provide any mechanisms to ensure the security of transactions, and also the ease of finding a desired product range is not very impressive. The second website is pyszneizdrowe.pl with the score of 62.13% points, and the two last websites, zdrowazywnosc.com.pl and biovert.pl, obtained about 50% of the maximum score. We observe a considerable dispersion between the first and the last website in the ranking, which in this case is almost 26 percentage points. A graphic illustration of the ranking scores is presented in Fig. 1.

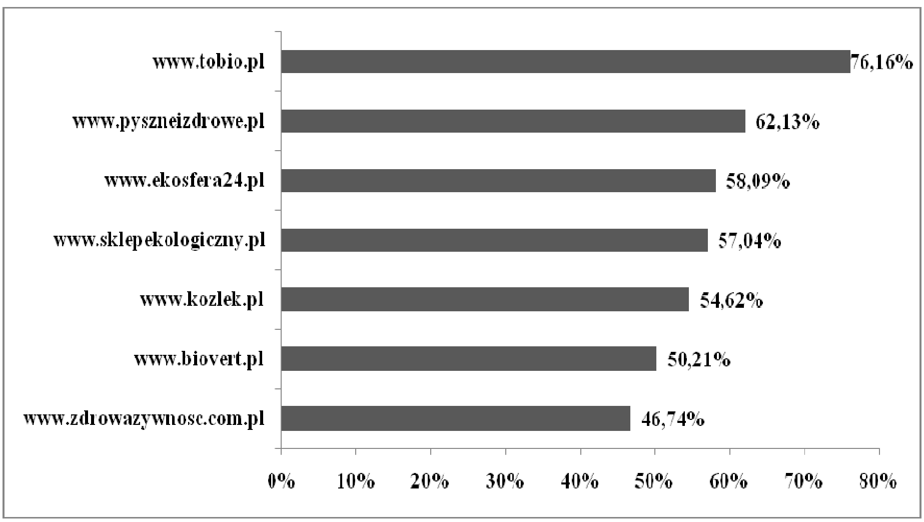


Fig. 1. Ranking of positions of examined online organic food shops

In particular categories and sets of categories we observe the following tendencies:

- the best score in the ranking: the ease of finding delivery terms – basically in all websites they could be found in at least one place in *Terms and Conditions;* internet users were aware of the fact and, consequently, the category of finding *Terms and Conditions* also takes the leading position,
- despite poorly (taking into account visualization and the applied technological solutions) designed websites, they are well positioned and they offer a wide range of organic food products,

- websites are equipped with efficient search applications and they offer various methods of payments,
- website's functionality and navigation receive high scores,
- in the case of security (which generally received the lowest scores) only one website offers an appropriate level of security when performing transactions,
- the attractiveness of prices, technological correctness and related content functionality leave much to be desired,
- marketing functions are perceived as insufficient – shops did not offer many promotions, the only manifestations of marketing operations were free deliveries of purchases exceeding a certain value and below a certain weight.

Generally, the experts assigned the highest scores for the criterion of the ease of finding the delivery terms and *Terms and Conditions of purchases and deliveries* (79%), availability in the Internet (75%) and the product range on offer (70%). In the next step, they evaluated search applications and methods of payment, user-friendliness and navigation of the website. In the last positions there were: security (14%) and attractive prices (43% - organic food is relatively expensive).

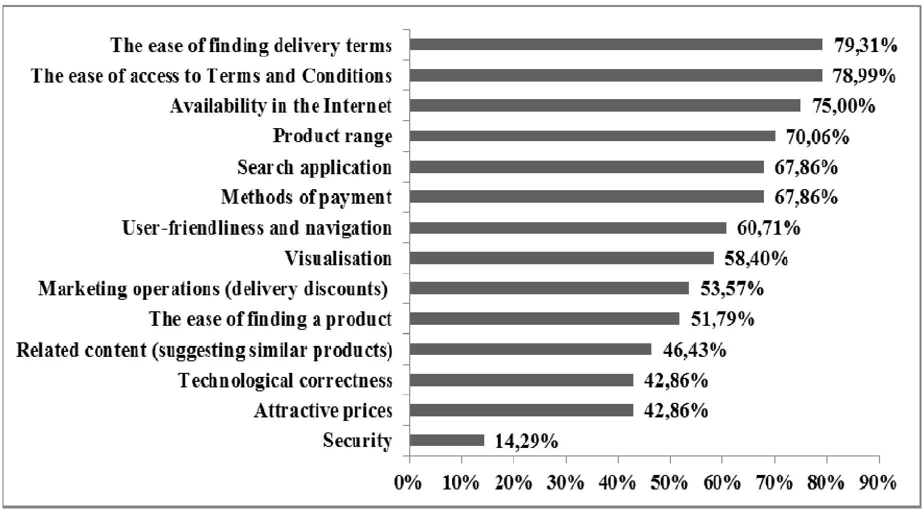


Fig. 2. The ranking of criteria fulfilment of the websites selling organic food

4 Analysis of Variants with a Preference Scale

In addition, the author carried out a number of experiments consisting in assigning preference points to sets of criteria. The first step in this phase of the study was a division of criteria into three groups mentioned previously: technical, functional and economic. The author applied five possible variants of assigning preference points for the criteria sets:

- technical (technical criteria - 60%, others - 20%),
- functional (functional criteria - 60%, others - 20%),
- economic (economic criteria - 60%, others - 20%),
- non-functional – because in this variant we distinguished the greatest number of functional criteria, economic and technical criteria received 40%, and functional 20%,
- user – in this variant a user assigned his/her own preference evaluation to all criteria, and subsequently, the preference evaluation, after the division into three criteria groups, was calculated in particular groups and amounted to: 55% for technical factors, 33% for functional factors and 12% for economic factors.

The results of calculations consisted in multiplying the total values of group evaluations by preference coefficients. The results are shown in the table below.

Table 10. Variants of website evaluations with marked preference scales according to economic indicators

	Economic	Technical	Functional	Non-functional	User
sklepekologiczny.pl	1.53	2.12	3.51	7.02	1.72
ekosfera24.pl	1.55	2.58	3.11	7.09	2.82
biovert.pl	1.71	1.93	2.91	5.94	1.55
kozlek.pl	1.75	2.15	2.87	6.17	1.99
zdrowazywnosc.com.pl	1.96	1.40	2.45	4.61	0.59
pyszneizdrowe.pl	1.98	2.39	3.53	7.30	2.03
tobio.pl	2.45	3.00	4.28	8.98	2.63

The application of various preference scales changed the order of particular websites. In the case of tobio.pl website, it appeared in the first position four times; it ranked second once (in the case of the user variant). In this variant it was clearly seen that a user chosen at random prefers good technical solutions of a website. Therefore, the first position in this case is taken by ekosfera24.pl (formerly ekosfera.pl), the shop which in the variant with preferences of technical solutions was in the second position. The next shop, pyszneizdrowe.pl, took a second position three times – in economic, non-functional and functional variants. In the case of this website technical imperfections appear to be balanced with high level of usability and economic factors. The worst website is zdrowazywnosc.com.pl, the website which occupies the last position in the ranking four times, and biovert.pl takes the penultimate place in the ranking three times. Zdrowazywnosc.com.pl is characterized by good results of fulfilment of economic criteria and biovert.pl received a high score with regard to its functionality. The next shop considered by researchers, ekologiczny.pl, does not stand out in the ranking and obtains the most average scores among the sector leaders. The evaluations of the preference scale are shown in the figure below.

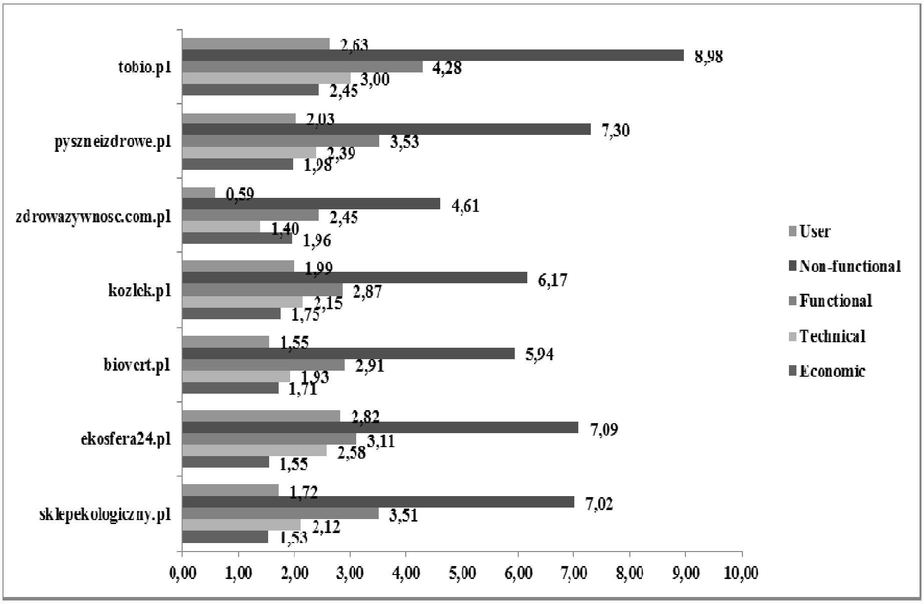


Fig. 3. Ranking of the fulfilment of evaluation criteria of shops selling organic food with the preference scale in five variants of relations of evaluation criteria groups

5 Instead of Conclusions

After summarizing the analyses of organic food websites we may observe that the dynamics of their development is very high. The comparison with very few earlier studies shows that the quality of websites is changing very quickly. Therefore, the analyses which are being carried out depict the situation at the moment of actual performing of the examination. Nevertheless, on the basis of the distinguished criteria, we may establish a specific ranking of selected shops. The tobio.pl shop may be used as a model for such type of websites, and it can be used as an example to follow when designing other websites which are to function in the sector. The examined internet shops are not large enough to be seen as specialist grocery vortals. Also, they do not represent particular producers. Due to the rapid development of organic food production it seems that there is no such possibility. Some websites are specific electronic distribution channels of shops that exist and function in economic reality. So, the question remains if these shops will be interested in moving all their operations into the realm of electronic commerce, and whether such a decision would be profitable for them from the economic point of view.

In the electronic economy developing in Poland, similarly to the whole sector of organic produce, there are still relatively few shops. However, in recent years the gap is being filled faster than in the case of sales of “regular” food. Nevertheless, the general impression is that in this segment – as analyses of the market may suggest – we observe a shortage of internet solutions and there emerge great possibilities for their future development.

References

1. Chmielarz, W.: Analysis and Evaluation of Selected Internet Groceries. In: Korczak, J., Dyczkowski, M., Dudycz, H. (eds.) Advanced Information Technologies for Management, AITM 2010, No 147, Prace Naukowe Uniwersytetu Ekonomicznego we Wrocławiu, Wrocław, pp. 61–72 (2010)
2. Chmielarz, W.: Quality Assessment of Selected Bookselling Websites; Polish Journal of Management Studies. In: Grabara, J., Kot, S. (eds.) Częstochowa University of Technology, vol. 1(1), pp. 129–147 (2010)
3. Evans, J.R., King, V.E.: Business-to-Business Marketing and World Wide Web: Planning Managing and Assesssing Web Sites. Industrial Marketing Management 28, 343–358 (1999)
4. `http://dziecisawazne.pl/rynek-zywnosci-ekologicznej-zwieksza-sie-o-okolo-30/`
5. `http://ec.europa.eu/agriculture/organic/home_pl`
6. `http://money.cnn.com/2007/05/14/news/economy/online_retailing`
7. `http://pl.wikipedia.org/wiki/%C5%BBywno%C5%9B%C4%87_ekologiczna`
8. `http://www.ijhar-s.gov.pl/raporty-o-ekologii.html`
9. `http://www.organic-world.net/yearbook-2011.html`
10. `http://www.zielonyszop.pl/?p=539`
11. Mateos, M.B., Mera, A.C., Gonzales, F.J., Lopez, O.R.: A New Web Assessment Index: Spanish Universities Analysis. Internet Research: Electronic Application and Policy 3(11), 226–234 (2001)
12. Migdadi, Y.K.: Quantitative Evaluation of the Internet Banking Service Encounter's Quality: Comparative Study between Jordan and the UK Retail Banks. Journal of Internet Banking and Commerce 13(2) (2008)
13. Miranda, F.J., Cortes, R., Barriuso, C.: Quantitative Evaluation of e-Banking Web Sites: an Empirical Study of Spanish Banks. The Electronic Journal Information Systems Evaluation 9(2), 73–82 (2006)
14. Selz, D., Schubert, P.: Web Assessment: A Model for the Evaluation and the Assessment of Successful Electronic Commerce Applications. Electronic Markets 7(3), 46–48 (1997)
15. Szynkiewicz, M.: Ocena użyteczności sklepów internetowych z żywnością ekologiczną (Evaluation of Usefulness of Online Shops with Organic Food), Faculty of Management, University of Warsaw, BA thesis, under the supervision of W. Chmielarz, Warsaw (2010)
16. Whiteley, D.: e-Commerce: Strategies, Technologies and Applications. McGraw Hill, United Kingdom (2000)

Evaluating the Application of Service-Oriented Auditing in the B2G Domain: A Case Study

Faiza Allah Bukhsh and Hans Weigand

Dept. Information Management, Tilburg University, P.O. Box 90153,
Tilburg, The Netherlands
{f.a.bukhsh,h.weigand}@uvt.nl

Abstract. In international trade, reliability, security and cost effective logistic chain management are very important challenges that can only be met by innovative usages of IT. The Extended Single Window project aims at a drastic reduction of physical inspections of goods in main ports by coordinated planning of government authorities, reliable transport to and from hinterland hubs and administrative cost reduction. Although quite some efficiency gains can be achieved already by improving the data logistics, a higher level of innovation requires a re-engineering of the control processes. Service-Oriented Auditing (SOAu)stands for the integrated application of smart auditing techniques in the framework of a Service-Oriented Architecture. In this article we evaluate the SOAu methodology on a B2G case study. From this study we have identified seven challenges: 1-simplifying declaration procedures 2-alignment of (multiple) inspections 3-data quality 4- increasing inspections efficiency 5-online monitoring 6-electronic data delivery 7-collaborative testing. We examine to what extent SOAu can provide solution directions for each challenge.

Keywords: Auditing, Customs Control, Service-Oriented Architecture.

1 Introduction

To reduce the burden on the front-office in government organizations the concept of e-government has been introduced. In e-government, most of the government functions and processes are carried out in the digital form over the Internet. Depending on the stage growth, different IT architectures can be used [11]. For the integration stage, a Service-Oriented Architecture (SOA) is very appropriate. SOA can be known as service based design principles for systems development and integration. Formally Open group and OASIS define SOA as "a paradigm for organizing and utilizing distributed capabilities that may be under the control of different ownership domains" [15][21].

Customs is an example of a government organization. Growing trade and increased security require new Customs controls. In parallel, governments would like to reduce the administrative burden. e-government provides the basis to introduce e-customs, where e-customs supports simplified paperless trade procedures, preventing potential security threats and counterfeit tax related frauds.

J. Grabis and M. Kirikova (Eds.): BIR 2011, LNBIP 90, pp. 281–295, 2011.

In addition, e-customs should ensure the interoperability with other e-custom systems within and outside the Europe [16]. Interaction of trading companies with e-custom give rise to the B2G concept [17].

Auditing in combination with SOA give rises to the SOAu concepts. The objective of Service Oriented Auditing (SOAu) is to implement continuous and online monitoring of services using SOA. The goal of our research is to evaluate the relevance of SOAu for e-customs following a design research methodology. As a first step, we have explored *requirements* on the use of SOAu in e-customs [2] from an analysis of the customs processes and challenges (*problem identification*). The *design* of a concrete SOAu solution has been worked out in [25]. The research objective of this paper is to take the next step, following the design research steps of Peffers [6] that is, *demonstration* of the effectiveness of the solution, by analyzing practical business problems in international transport and examining to what extent SOAu can address these problems. We do not see this as a complete evaluation, but as an important part of it.

The paper is organized as follows. Section 2 gives an overview of the basic concepts and describes the Extended Single Window (ESW) project of which our research is a part. It also gives a brief description of the demands and problems of our industrial project partners. Section 3 summarizes the SOAu methodology. Different audit types and audit service configurations are distinguished. Section 4 evaluates the presented approach on a German based Food&Beverage company. Section 5 closes with the main conclusions and directions for future research.

2 Background

ESW [5] project's main focus is on creating a faster, safer and more reliable international flow of goods through the smoother and more efficient processing of import, export and customs procedures. Its main objective is to strengthen the role of the Netherlands as the logistics gateway for Europe and to contribute to added value in supply chain coordination.

2.1 Basic Concepts

Some basic concepts in terms of customs and trade procedures are the following:

Supply Chain Management (SCM): SCM is the movement of materials as they flow from their source to the end customer. Formally we can define a supply chain as a network of facilities and distribution options that performs the functions of procurement of materials; transformation of these materials into intermediate and finished products; and distribution of these finished products to customers [9]. Government authorities such as Customs, police, and quality inspection agencies intervene in the supply chain. Confronted with new security challenges, Customs cannot focus anymore on a single actor, but should extend its scope to the whole supply chain [10].

Modernized Customs Code: International trade in EU provides the base for the revision in border management regulations and custom procedures. On the one hand it is the objective to simplify Customs procedures, whereas on the other hand security has to be safeguarded. In the first phase, the Authorized Economic Operator (AEO) certificate for trusted traders [8] has been introduced. In the second phase, Customs is being revised in accordance with the Modernized Customs Code (MCC). Single Window (SW), Single Authorization for Simplified Procedures (SASP), and Self Assessment enable centralized control of goods flows in the EU. The Single Window concept is an example of coordinated border management. AEO, SASP, SW and MCC are milestones towards security, efficiency and reliability challenges in today's trade procedures. For import/export each company has to provide piles of documents to several government authorities which implies high costs for the business. However, lots of business data are already available within the supply chains. Cost reduction can be obtained by re-using these business data for different government authorities. Furthermore, many businesses have implemented their own risk management systems for resilient and reliable supply chain management. Again, costs can be saved by building on these systems.

Single Window business perspective: The SW concept is based on data re-usability by all government authorities for all types of goods movements and can be defined as " a facility that allows parties involved in trade and transport to lodge standardized information and documents with a single entry point to fulfill all import, export, and transit-related regulatory requirements. If information is electronic then individual data elements should only be submitted once" [22]. Two implementation scenarios for SW are: (i) Inter operable government portals with a transaction based declaration approach. Its objective is to re-use information between authorities based on one declaration by business and it follows the data push principle. It increases the efficiency of border management by alignment of inspection planning of different authorities. Currently, each inspection leads to additional handling, waiting times, and thus additional costs. (ii) SW implemented in business processes. In this scenario business processes of logistic actors gather all relevant information, including physical cargo/container tracking for instance GPS technology. This information is made available to the government authorities like Customs. This scenario encourages the data pull principle. It also allows these authorities to track goods movements across borders.

2.2 Extended Single Window and Its Stakeholders

The objective of ESW is to increase supply chain reliability, reduce administrative costs for all stakeholders and improve effectiveness and efficiency of government authorities. It considers various inspection and law enforcement requirements such as security, health, economic, environmental and fiscal controls. Multi directional data flow is possible by end-to-end supply chain integrity [10]. According to MCC, ESW covers incoming and outgoing logistic flows, including integration with previous (outgoing goods for instance preceded by export) and subsequent procedures (incoming goods for instance followed by transit).

EWS project is a consortium that comprises knowledge institutions (TNO, Delft University of Technology, University of Tilburg and the Fontys and NHTV Universities of Applied Science), the Dutch main ports and their community systems (Rotterdam Port Authority, Port of Amsterdam, Schiphol, Portbase and Cargonaut), logistics hinterland regions (NV Regio Venlo), interest groups and industry associations as well as international operators. Dutch Customs is involved in the innovation. The main stakeholders of ESW can be divided into three categories: government authorities, international operators and facilitating third parties between government and international operators. International operators are diverse in the nature of business. Therefore ESW can cover a broad range of trade requirements. A short description of some of the international operators (IO) is as follows:

1 -IO-1 is a large international supplier for blends, compounds, fruit preparations, fruit juice concentrates, flavors and colors, emulsions and ingredient systems. This company does business in more than 130 countries and has got around 2,500 employees.
2 -IO-2 is mainly selling foodstuff and personal care items via independent distributors. Typically they do direct sales to end consumers in a Tupperware type of marketing sales. This company is active in 74 countries.
3 -IO-3 is the distributor of electronic components and has one of the largest and most modern component warehouses in Europe. For the delivery to 35.000 European customers there is a need to manage different documents (invoices, order confirmations, delivery notes, etc.).
4 -IO-4 is a single pharmacist and they develop and produce printers for the professional market and media suppliers. They operate in high-end printing technology for the office sector, banking sector and high-volume sector.
5 -IO-5 is the toy (e.g. dolls) and games (board games, video games) company. Its domestic business is in USA and other international businesses are in Australia, New Zealand, Canada, Europe, Asia and South America.

IO 1-4 are AEO certified while IO-5 is currently in the process of acquiring an AEO certification. The diverse nature of these consulted industrial partners [28] will help us in identifying potential improvements for coordinated inspection by customs and other authorities in line with global and EU developments as laid down in for instance the MCC. Although all international trade processes are considered, the primary focus is on export.

For efficiency and reliability each company have their built in control, controls such as Langdon support e-declarations with numerous countries, e-card system, digital declaration etc help them to communicate with custom. These built in controls are usually ready to be connected with the SOA based Event Service Bus that is envisioned as pilot in ESW. Many IT applications/services in the consulted business companies are offered by the SOA. This implies that a SOAu approach is certainly applicable.

3 Service Oriented Auditing Methodology

Service Orientation is the emerging IT standard whose global openness, design for agility, dynamic orchestration and single-source approach offer new opportunities in trade and logistics. The Fig. 1 shows the context of the proposed

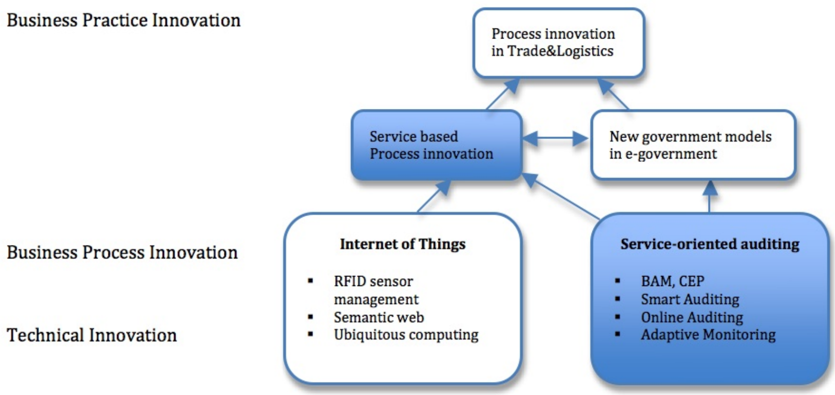


Fig. 1. SOAu research framework

research to achieve SOAu. The ultimate objective is process innovation in trade and logistics, like an ESW. To support this innovation, the research aims to explore process innovation (at the concept level): service-based process innovation in the transport domain, new governance models in the administrative domain, and especially the interplay between these two [1]. The process innovations that we foresee draw on already existing and still to be developed technical innovations that we group together under the label of SOAu.

SOAu aims at the use of service-oriented technology to support auditing processes, in particular continuous and online monitoring. An audit module based on SOAu monitoring solution has been defined by Weigand and Bukhsh [25]. In this audit module we have considered that business services generate business events (including service request and service response events) and all of them can be published on the enterprise information service bus (EISB). With respect to continuous monitoring (CM) service two categories of events have been studied (i) operational events (ii) economic events - using the REA business ontology [13]. In the audit module a fault list is being generated which can be forwarded immediately to the stakeholder (Company / Government/ Concerned Authorities (push) or made available online (pull)).

Traditionally, a distinction is made between internal and external auditing that do not exclude but complement each other. In SOAu we have found a new classification to be highly relevant at least for the applicatoin area of e-customs. First of all, we divide the government and company audit relationship into two categories: (i) uncoordinated: in which one company is audited by only one government authority at a time; (ii) coordinated: when many collaborating companies are audited by many government authorities in a coordinated manner. The auditing itself may be of four types.

3.1 Uncoordinated Auditing

Uncoordinated means that one government authority audits one company at a time. There exists direct communication between the company (e.g. trading company) and government authority (e.g. custom). In this category, there are again two subcategories, depending on the audit subject: (i) company audits itself, reporting to the government. According to Starreveld et al [20] organizations need internal control measures, including organizational rules and control activities. The purpose of auditing the internal controls by the company is twofold, one is to attract the investors and other is to show the transparency of the company to the government; (ii) government audits the company. This audit is of detective and corrective nature. The government wants to check the status of the organization/company's declarations and trustworthiness. Government authorities or shareholders or investors usually perform this type of audit. They audit the assets, controls, declarations and all the stability related matters.

When an organization provides (atomic and composite) services to users, a distinction can be made between operational services and control services. The operational service [3] describes a specific type of business service performed by human or machine. Control (management) services [26] realize business control of the operational services.

Now four types of audit service can be distinguished, depending on who is performing the audit (audit subject) and what is the primary focus of the audit (audit object):

Type I: The company audits its operational processes. Traditionally, this type is not feasible as the government is not willing to leave the auditing responsibility to the company. However, with new technology we can envision a (certified) audit module that supports a continuous monitoring service of the company's operational services and can detect and immune any potential operational issues. Such an audit module may be acceptable to Customs.

Type II: Companies arrange the audit activity especially auditing the control services for itself. The self-assessment can replace costly governmental controls, as in the case of custom procedures. For this purpose e-custom provide a standard known as AEO certificate [8]. An audit module audits the company's control services in order to ensure compliance to the AEO standard.

Type III: The government audits the operational services. This case has two scenarios (i) government physically audits the operations and operational services. This is the traditional way of working, Evidently, this is a labor-intensive process. (ii) Government audits the operational processes by using advanced IT such as automated scanning and smart auditing techniques such as proposed by van Aalst et al [23].

Type IV: The government remotely monitors and evaluates the control system. This variant may benefit both the company and the customs. It requires an even higher level of "being in control" then in the case of type II, In other words, the internal control system must be highly formalized and automated. The costs that this brings for the company are compensated by the fact that (manual) self-assessments are no longer needed: the company only needs to provide access to the control services, via some service interface.

3.2 Coordinated Auditing

When multiple government authorities audit multiple companies require new controls. In parallel government would like to reduce administrative burden. In this context, coordination emerges as a separate service which is the base of the coordinated auditing. Coordination services can be defined as services supporting an exchange process (a set of events) for a good or a service [27]. Within this process, a distinction can be made between core services - the transfer of goods, services or money - and coordination services that support the process and manage the dependencies between activities [12].

Suppose there are n government authorities auditing a certain transport in which m companies are involved. In total there may be n.m combinations of audits. To make this process more efficient, the concept of trusted third part may be adopted, that is, audit by third party. The third party audits the company, combining the different requirements from different government agencies, and takes a chain perspective rather than focusing on one company only. Not only efficiency, but also effectiveness can be increased in this way, as it allows cross-validation of the provided data. The type of audit is of detective, corrective and preventive nature.

For coordinated auditing, three coordination types can be distinguished, from minimal to maximal (Table 1).

Table 1. Coordination types

Coordination Type	Coordination Level	Coordination By
Type A	Minimal	Company
Type B	Partial	Government
Type C	Maximal	Third Party

4 B2G Case Study

IO-1 is a German based large international supplier for blends, compounds, fruit preparations, fruit juice concentrates, flavors and colors, emulsions and ingredient systems. Its headquarter is located in Darmstadt Germany. One of those centers is located in Oosterhout, the Netherlands which is the largest sales and production location. IO-1 has business in more than 130 countries. IO-1 considers aspects like food safety and microbiology, Sensory & Consumer Science, Trend-Monitoring, Market Intelligence, packaging technology, developing innovative product applications and Sourcing and Supply chain Management for its business. It takes on different roles in the business network; this really depends on what the demands from the market are. For instance, when the market demands the goods to be manufactured more, IO-1 can act on this demand and fabricate the desired products. In such cases IO-1 take on the role as an intermediate. In some scenarios IO-1 work as an intermediary between the different tiers in a network. Coleman et al [19] emphasize that networks can be divided into three categories, namely a stable network, an internal network and a dynamic network. IO-1 business network is a form of Dynamic network. A dynamic network is built on different actors in the value chain who are coupled in a contract format. In IO-1 actors are coupled on long term as well as short term (i.e actors will make a contract for a single transaction and will decouple again after completion of the transaction) contract. According to Schmid three main phases need to be completed to make a successful transaction in a dynamic business network [18]. These phases include the information phase, negotiation phase and the settlement phase. In IO-1 all of these phases are supported electronically.

4.1 Supply Chain of IO-1

A supply chain describes every actor in the chain who contributes to a product [4] [14]. For instance, if a company produces fruit juice, then its supply chain will include the units where the raw materials are made into concentrates, the supplier who produces the added sugar, and the supplier who produces the product packages and so on. As an example, supply chain model of IO-1 has been studied in depth. This model is based on the chain that is constructed on the supply of Concentrates. IO-1 has access to raw materials, so they have perfect control on the quality aspects on the production of raw materials. However, IO-1 does not own the production fields for growing these raw materials. This supply chain is chosen because it complies with the following characteristics: (i) it shows the whole chain from raw materials till end-product, (ii) it gives insight in the international trade (from Brazil to the Netherlands) and (iii) gives insight in trade between countries from the EU (the Netherlands and Germany). This supply chain will be used as basis for the case study. Fig. 2 describes the supply chain model which starts with the production of raw materials (fruit). The farmer, who produces fruits, sells them to a production facility or sells these directly to the company. Most of the fruits that are used for concentrates are produced in Brazil; The production facility in Brazil is not owned by IO-1 but is an independent actor within the supply chain. This production facility manufactures concentrates from raw materials. After finishing the semi-product, these materials are shipped to the IO-1 distribution facility located in Brazil. This distribution facility produces bulk materials from the collected semi-products. These bulk materials are then prepared for transport from Brazil to the Netherlands. IO-1 hires an independent subcontractor who is responsible for the shipment of goods from Brazil

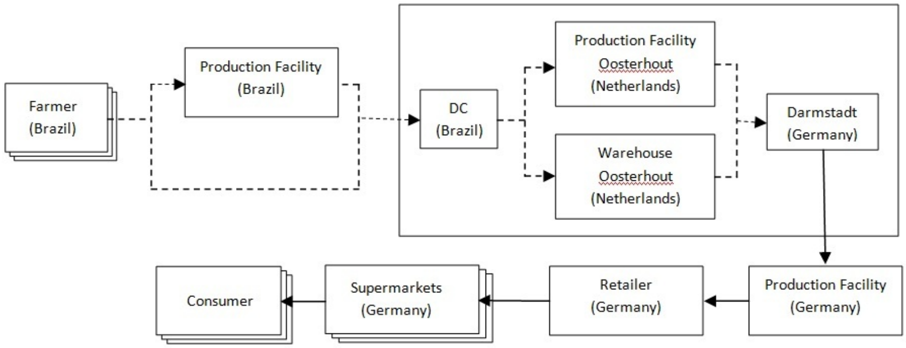


Fig. 2. Supply chain of IO-1

to the Netherlands. When the semi-products arrive in IO-1 Oosterhout (The Netherlands) these products can either be fabricated more in the production facility or can be stored in a bonded warehouse. Most of the concentrates that are imported from Brazil are imported in the Netherlands and will be processed for shipment to the IO-1 facility in Darmstadt (Germany). About 90% over the imported concentrates, who have destination Darmstadt, are being declared for customs in the Netherlands. So Darmstadt does not have to declare them to customs when they arrive in their facility. From this IO-1 facility in Germany the products are shipped to the retailers. From these retailers the products are moved to the supermarkets and finally to the consumer.

4.2 Built-in Controls of IO-1

The E-card system, that is used by IO-1 for all their documents related to customs authorities, has a wide range of built-in control functions. For instance, it checks the completeness, consistency, accuracy and reliability of the inserted data. This system is also protected with authentication security, so only authorized personal can operate this system. The E-card system also checks if the Movement Reference Number (MRN) that is inserted corresponds with the right shipment. It also has currency converter for calculating the right currency. It is capable of providing written reports and positioned to deliver the useful information that is subject to audit, mostly an import or export event. This report is provided to the external client, mostly the customs authorities (the "independent auditor"). The E-card system interacts with the Transit system from the customs authorities. The whole process of delivering documents electronically through the E-card systems by IO-1 is based on a "push" model. When containers arrive in the facility in Oosterhout only qualified personal are authorized to open and check the containers. They do a thorough check, in a secured room, for the presence of gas in the containers. These gassed containers can be hazardous for any person who opens it and can cause severe physical issues. Only containers who have a certified gas free statement are qualified as safe, every other container is checked for the possible presence of gas. According to publications from VROM [24] in 10% of all checked containers in the Port in Rotterdam the presence of gas was found. After the completion of the gas check the personal do quality and completeness tests on the cargo in these containers. They check if the cargo does comply with the wide range of standards (9001:2008, ISO 22000) that apply for these containers. IO-1 is AEO certified with certificate F; this means that they have the customs authorities simplifications certificate combined with the security certificate. In order to obtain this certificate they did a self-assessment and have clearly defined the outline of different departments like production, logistical and reception.

4.3 Problem Description

The problem description is based on structured interviews [29] with in-company experts on supply chain management and customs procedures. In these interviews different problems were identified that we categorized into 7 challenges.

Challenge 1 : Simplifying declaration procedures; In our case study, cargo is imported by IO-1 from Brazil. This cargo will be processed for storage, further production or shipment by the facility in Oosterhout (the Netherlands). Two different approaches in declaration are identified. Both of these approaches entail goods that are declared in Oosterhout, even when they have destination Germany, they will be declared in Oosterhout. The first procedure is the declaration process for goods with destination Netherlands. These goods will be stored in a bonded warehouse where they will await for further processing or shipment. The second approach is the declaration of goods with destination Germany that are transported through Oosterhout. These goods are bought by the IO-1 facility in Darmstadt (Germany) but are declared in the facility in Oosterhout. This process has got a longer handling time and has more interactions. Unification of these two approaches will lower the costs and processing time.

Challenge 2 : Alignment of (multiple) inspections; If a container gets picked out in the port of Rotterdam, for inspection or a scan procedure, it takes a long period, usually ten days, before a container gets released. This delay occurs because there is a substantial amount of time between selection and transportation process and the actual inspection. The selection procedure and the scan procedure should be better aligned, which will result in lower processing and transporting time on scan.

Challenge 3 : Data quality; Sometimes a wrong serial number of a container, the MRN, has been given to customs employee for inspection. This employee then selects and inspects the wrong container. By mistake the wrong container held back for inspection so the right container can, for instance, already be on its way to the next destination. Of course this results in delay of the inspection.

Challenge 4 : Increasing inspections efficiency; Through a pronouncement the customs office announces that an inspection will take place on site, for instance at the IO-1 facility in Oosterhout. When this announcement has been given all progress with that particular shipment must be stopped. The actual inspection however takes place 4 till 5 hours later. So there is certain amount of time and work lost, because inspections that take place on site, are not aligned correctly with their announcement.

Challenge 5 : Online monitoring; Now the procedure for declaring goods is through the use of the E-card and Transit system. However, customs office requests all the declarations of a particular month and wants them burned on CD-ROM and delivered through the mail. IO-1 would prefer to send these monthly declarations digitally or publish all the documents, related to customs authorities, in an online database from where custom can easily fetch the requested documents for further processing, monitoring and auditing.

Challenge 6 : Electronic data delivery; A lot of certificates that are needed, from embassies and public authorities from foreign countries, are still processed through paper form. The request and process procedure of these documents is costly and usually takes up a lot of time.

Challenge 7 : Collaborative testing; When the customs authorities in the Netherlands publish a system update or put a new information system online, there is no possibility to test this new system or system update first. A test environment is needed to eliminate errors and bugs before the system goes Live (online).

4.4 Problem Analysis Based on SOAu Methodology

IO-1 is AEO certified with certificate F, so they have the benefit of fewer inspections and the possibility of doing their customs declaration through electronic transfers. The E-card system facilitates in the electronic transfer of customs related issues, so it facilitates automated export/import management. Different government authorities ask for different information to have an inspection on the ports. In parallel IO-1 has to provide information to the logistic companies. Fig. 3 shows how the IO-1 works currently. Built-in controls in the 'e-card' system

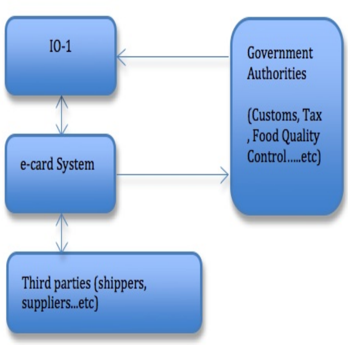


Fig. 3. Current State of the IO-1

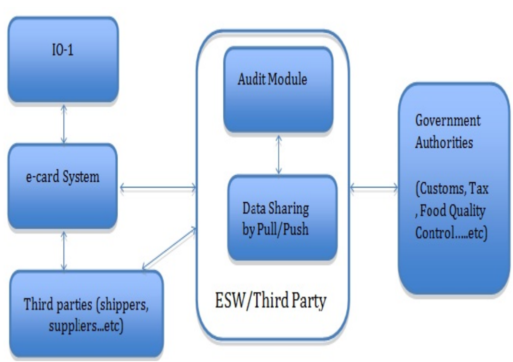


Fig. 4. Expected State after application of Extended Single Window for IO-1

prepare and provide all the information needed for the government authorities and for the outsourcing agencies. In the IO-1 supply chain we can see that it outsources its logistic process to trusted logistic companies. Which gives rises to the concept of third-party. Therefore we can call that IO-1 have coordination service. Based on the SOAu methodology if custom audits IO-1 then there exist partial coordination relationship. Which means one government authority is auditing many coordinating companies in parallel. IO-1 is audited by many government authorities during import and export such as customs authorities, food quality control authorities, excise authorities and many more. This gives rise to the coordination among the government authorities to save time and effort. In case of coordinated auditing multiple cooperating government authorities coordinate/audit/communicate with multiple coordinating companies. Based on SOAu methodology If we look in the (Table 1), IO-1 fits well in the division of Type B where there is less coordination between government authorities. However different work flows/processes must also be changed to eventually be able to improve the Type B into Type C. Type C implements the Extended Single Window concept fully. By using smart audit modules, trusted third party, ESW concepts and SOAu methodology we can improve the communication between the IO-1 and government authorities. Fig. 4 shows our proposed scenario, Fig. 4 shows that all the required information and data is kept by the "ESW/Third party" in parallel with audit module. This third party manages all the requirements by the government authorities as well as by the companies and outsourcing parties (shippers, suppliers, logistic companies etc) by using ESW concept. We will evaluate the validity of the proposed solution direction on the IO-1's challenges,

Solution 1 : Simplifying declaration procedures; From the company's perspective, different declaration procedures cause extra work load but from the custom's perspective it is necessary for securing the import/export procedures for different companies/countries. SASP provide a way to declare the goods once in the EU and then use that declaration for further movement of goods [7]. To be able to use the SASP facility companies should be AEO certified, Type II provide us a way to get an AEO certification. Thus we can say that in case of uncoordinated auditing if a company satisfies Type II & Type III then problem of declaration procedures can be minimized. In case of Coordinated auditing Type C form (Table 1) enhances maximum coordination between government authorities and between companies and provides a viable solution to these conflicting goals.

Solution 2 : Alignment of (multiple) inspections; There exist many checks for the IO-1 during import/export at the border. In case of uncoordinated auditing, only aligning the audit of operational services and control services can reduce the administrative burden as introduced by the Type III and IV. In coordinated auditing maximum coordination between the government authorities is needed which is ensured by Type B. If inspection authorities fetch the data from ESW/Third party window, the processes and checks can be aligned. This will result in lower waiting times and more efficiency for the operators and government authorities.

Solution 3 : Data quality; Once the information is directly fetched by the ESW/Third party from the E-card (the operational system) and push it to the government authorities. Pulling and pushing of data will be as a stream of data so there will be less chance of having a wrong MNR number. Mistakes usually happen while copying data from one format to another.

Solution 4 : Increasing inspections efficiency; Companies that are more in control like Type III, IV, B and C provide maximum information to the ESW/Third party for inspection purpose. Based on this information scheduling the inspections can be more accurate. When the ESW/Third party extends its scope to earlier steps in the supply chain, more data is available and at a much earlier time. Moreover, by fetching data directly from the operational systems of the operator, it is easier to plan multiple inspection in less time.

Solution 5 : Online monitoring; For this problem a separate module exists in the ESW/Third party. This audit module keeps record of all the processes in the form of published events. These events will be audited based on some standards and a fault list will be produced. The audit module is applicable to coordinated and uncoordinated auditing. It also provides the supplementary knowledge for solving challenge 2 and 4 as well.

Solution 6 : Electronic data delivery; Certificates needed by embassies and public authorities from foreign countries can be accessed form the ESW/Third party. Data access from ESW/Third party is determined with respect to the level of authority and demand/needs of the authority.

Solution 7 : Collaborative testing; It is not reasonable to expect a solution to this problem from the government agencies, as it is not their core task. However, an ESW/Third Party could providing testing as as an additional service for the companies. Offering this service to multiple operators has an economy of scale advantage. Note that this is not an exclusive advantage, as it applies to any IT solution provider in this area

5 Conclusion

SOA is a de facto standard architecture based on the concept of services. SOAu - Smart auditing techniques in combination with SOA - provides a massive potential for innovation. A methodology for SOAu has been developed in [25]. The methodology is being evaluated in ESW perspective together with the ESW industrial partners. One of the international operators has been selected as a G2B case study. We have described the current state of the business of IO-1, identified concrete problems and generalized them into more general challenges. Although more case studies are needed to consolidate this list of challenges, it already contains important and specific directions for further research and development. We have used the list of challenges to evaluate the SOAu methodology. We observe that SOAu provides concrete solution directions for each challenge. Hence we can conclude that SOAu is a relevant solution direction in e-customs, at least from a business perspective.

The ESW/Third party concept is one of the components of SOAu. To the best of our knowledge, it is a new concept. In subsequent research, we hope to develop it further in multiple design cycles, drawing on existing and to be developed smart auditing techniques. Design research methods and case studies with international operators will be used for further evaluation.

Acknowledgments. This research is supported partially by the DINALOG[1] project Extended Single Window. We are thankful to Wouter Dusee for conducting interview with IO-1.

References

1. Baida, Z., Rukanova, B., Liu, J., Tan, Y.: Rethinking EU trade procedures–the beer living lab (2007)
2. Bukhsh, F.A., Weigand, H.: E-Government controls in service-oriented auditing perspective: Beyond Single Window. In: IFIP EGOV, Workshop Delft, the Netherlands (submitted, 2011)
3. Chen, R., Blaschuk, W.: Defining Operational Services:How SOA Can Help Realize a Role-Based Organization. SOA Magazine, Issue XIV (January 2008), http://www.soamag.com/I14/0108-3.php
4. Christopher, M.: Logistics and supply chain management: creating value-added networks. Pearson Education, London (2005)
5. DINALOG. Extended single window-information gateway to Europe, http://www.dinalog.nl/institute/projects/research-development-projects/extended-single-window-information-gateway-to-europe/271 (2010)
6. Ellis, T.J., Levy, Y.: A guide for novice researchers: Design and development research methods. In: Proceedings of Informing Science & IT Education Conference, InSITE 2010 (2010)
7. European Union. Single authorisation for simplified procedures (SEA) (2006), http://ec.europa.eu/taxation_customs/resources/documents/customs/procedural_aspects/general/centralised_clearance/1284_rev1-en-sea_guidelines.pdf.
8. European Union. European custom code (2008), http://ec.europa.eu/taxation_customs/customs/policy_issues/customs_security/aeo/index_en.htm (retrieved on January 11,2011)
9. Handfield, R., Nichols, E., Ernest, L.: Introduction to supply chain management. Prentice Hall, Englewood Cliffs (1999)
10. Hesketh, D.: Seamless electronic data and logistics pipelines shift focus from import declarations to start of commercial transaction. World Customs Journal 3(1), 27–32 (2009)
11. Janssen, M., Veenstra, A.F.V.: Stages of growth in e-government: An architectural approach. The Electoronic Journal of e-government 3, 193–200 (2005)
12. Malone, T., Crowston, K.: The interdisciplinary study of coordination. ACM Computing Surveys (CSUR) 26(1), 87–119 (1994)
13. McCarthy, W.: The REA accounting model: A generalized framework for accounting systems in a shared data environment. The Accounting Review 57(3), 554–578

[1] www.dinalog.nl

14. Mentzer, J., DeWitt, W., Keebler, J., Min, S., Nix, N., Smith, C., Zacharia, Z.: Defining supply chain management. Journal of Business Logistics 22(2), 1–26 (2001)
15. OASIS. OASIS SOA reference model (2011), http://www.oasis-open.org (retrieved on March 20,2011)
16. Razmerita, L., Bjorn-Andersen, N.: Towards ubiquitous e-custom services. In: Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence, WI 2007, pp. 833–837. IEEE Computer Society, Los Alamitos (2007)
17. Reddick, C.: A two-stage model of e-government growth: Theories and empirical evidence for US cities. Government Information Quarterly 21(1), 51–64 (2004)
18. Schmid, B.: Electronic Mall: Banking und Shopping in globalen Netzen. Teubner, Stuttgart (1995)
19. Snow, C., Miles, R., Coleman Jr., H.: Managing 21st century network organizations. Organizational Dynamics 20(3), 5–20 (1992)
20. Starreveld, R., de Mare, H., Joels, E.: Bestuurlijke informatieverzorging: Algemene grondslagen. Samsom (1985)
21. The Open Group. SOA work group member site (2011), https://www.opengroup.org/projects/soa/ (retrieved on March 20, 2011)
22. UN/CEFACT. The single window concept. Technical Report ECE/TRADE/324, International Trade Procedures Working Group (ITPWG/TBG15) of UN/CEFACT (2009), http://unpan1.un.org/intradoc/groups/public/documents/UNECE/UNPAN019892.pdf
23. Van Der Aalst, W., Van Hee, K., Van Der Werf, J., Kumar, A., Verdonk, M.: Conceptual model for online auditing. Decision Support Systems 50(3), 636–647 (2011)
24. VROM. Resultaten actie tegengas(2009), http://www.rijksoverheid.nl/bestanden/documenten-en-publicaties/kamerstukken/2010/02/12/resultaten-actie-tegengas-2009-rapport/rb2010004003aeindrapportageuitvoeringmotiepoppeboelhouwer.pdf
25. Weigand, H., Bukhsh, F.A.: Supporting customs controls by means of service-oriented auditing. manuscript Department of Information Management, University of Tilburg (2011)
26. Weigand, H., Johannesson, P., Andersson, B., Jayasinghe Arachchige, J., Bergholtz, M.: Management services – A framework for design. In: Mouratidis, H., Rolland, C. (eds.) CAiSE 2011. LNCS, vol. 6741, pp. 582–596. Springer, Heidelberg (2011)
27. Weigand, H., Johannesson, P., Andersson, B., Bergholtz, M., Arachchige, J.: Closing the user-centric service coordination cycle by means of coordination services. LNBIP, vol. 72, pp. 267–282 (2011)
28. Zarkasyi, S.W.: International audit techniques traditional versus progressive approach. Jurnal Ekonomi dan Bisnis Terapan 2 (2006)
29. United States Office of Personnel Management, Structured Interviews: A Practical Guide (September 2008)

Cloud Computing and Economic Growth in the Baltic Sea Region Countries

Andrzej Kobyliński and Marcin Michalski

Warsaw School of Economics,
Department of Business Informatics,
Al. Niepodległości 162, 02-554 Warsaw, Poland
Andrzej.Kobylinski@sgh.waw.pl, mmichalskipm@gmail.com

Abstract. For small and medium-size enterprises cloud computing (CC) delivers ICT resources flexibly, depending on required needs. Analysis of data from the Baltic Sea Region countries indicates that the prevalence of the CC model can contribute to higher economic growth. At the same time SMEs are asking many questions whose answers we do not know so far, but in the near future we will have to answer them.

Keywords: cloud computing, economic growth, Baltic Sea Region countries, SME.

1 Introduction

Endogenous growth theories (e.g. [1]), which arose in the late 1980's, explained that technological progress has a significant impact on economic growth. Development of technologies is supported by competitive markets, in which the main competitors are small and medium-sized enterprises (SMEs), generating the highest percentage of GDP and employing proportionally the largest number of workers [2]. Cloud computing (CC), as a new model of delivery and use of information and communication technology (ICT) resources, allows SMEs to gain inexpensive access to cheaper infrastructure and higher-performance (in their newest versions) than before. Until recently, access to such ICT resources was expensive and reserved for large companies. The flexibility entailed by CC allows ICT resources to be quickly and inexpensively increased or decreased, depending on required needs, facilitating faster response to market requirements, as well as the transfer of companies expenditures from capital to operational expenditures and the elimination of entry barriers [3]. The aim of this paper is to analyze the impact of investments in cloud computing on economic growth in the Baltic Sea Region countries.

2 Literature Review

2.1 Information and Communication Technology

Just as energy, ICTs have become a common good necessary in our personal lives or business activities, and thus an important component influencing the development of

J. Grabis and M. Kirikova (Eds.): BIR 2011, LNBIP 90, pp. 296–303, 2011.

economies, which as a tool to support business passed metamorphosis (e.g. [3]). ICT has ceased to be only a factor of increasing efficiency by reducing costs, but is now used as a tool for supporting innovation and creating new business models, in particular new services or ways of cooperation [4]. The decrease in domination of vertical integration as the classical industrial model as a result of ICT, allows companies to focus on their core competencies and business process outsourcing of non-core activities [5]. ICT has become an essential part of business investment in each country and Carr [6] estimates that approximately 50% of companies' expenditures (i.e. investment, but also expenditures on maintenance of existing infrastructure) are spent on ICT.

ICT contributes to economic growth in two ways: through the production of ICT products, as well as through the adaptation of ICT in the economy via investment and expenses. Jorgenson and Vu [7] argue that investments in tangible assets, including hardware and software were the main driver of economic growth in the world after 1995 (another factor was more effective use of labor). In the years 1993-2005 investment in ICT led to 0.35-0.90% GDP growth in most OECD countries (of which one third is expenditure on software) [8]. Vu [9] shows that the accumulation of ICT capital significantly determines changes in the growth of production in the countries, both developed and developing. In addition, for a given level of the labor and capital growth, a higher level of ICT investment per capita enables to achieve higher growth production rates. Nasab and Aghaei provide data for OPEC countries [10], arguing that the increase in expenditure on ICT by 10% would result in 0.2% growth of GDP per capita.

2.2 Cloud Computing

Cloud Computing (CC) is considered as a new model of delivery of ICT whereby, on request, the user gains access via the Internet to computer resources (memory, network, servers, computing power, applications and development platforms), quickly and flexibly, with minimal interaction with the provider. In reality cloud computing is not a new trend, but the aggregate new name for the number of trends existing in IT world for many years: computing on-demand, utility computing, application service providing (ASP) [11]. According to the U.S. National Institute of Standards and Technology (NIST) [12], there are three CC types: Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS), and four methods of implementation: private, public, social and hybrid clouds. Currently, CC is in the initial phase of the market formation, with relatively few suppliers as: Amazon (IaaS), Microsoft (PaaS), Google and Facebook. Amazon was a pioneer in launching Amazon Elastic Compute Cloud (Amazon EC2), huge server farms that companies can rent [13]. Microsoft's Azure platform for developers enables application creation and hosting [14] as is Google App Engine [15]. Facebook too has released a platform where software developers can create applications hosted on their servers, operating on and cooperating with the social portal [16].

The main benefit of CC is the transfer of capital expenditure (CAPEX) to operating expenditure (OPEX) which depend on the actual demand for ICT resources, removing an entry barriers especially for SMEs, creating employment, increasing production and reducing the margin by increasing competition [3]. According to the Boston Consulting Group (BCG), Cloud Computing is offering three levels of benefits [4]:

- Utility level. The beneficiary at this level is the CIO due to capital costs becoming operational costs; more predictable costs; shorter time to the implementation of the new applications; guaranteed service, support and maintenance by the provider, reducing business operating costs.
- Process transformation level. The beneficiary at this level is the enterprise as a whole due to improved business processes through the use of applications hosted by the supplier, accessible from any device connected to the Internet.
- Business-model-innovation level. The beneficiary at this level is the business ecosystems. For example, the value chain in health care consists of pharmaceutical companies, insurers, doctors, hospitals and patients, but patient data are inaccessible along the chain. Gradually this may change. Quest Diagnostics [17], for example, provides access to laboratory tests results via the SaaS applications Google Health [18] and Microsoft HealthVault [19].

Brynjolfsson, Hofmann and Jordan [20] believe that as CC is becoming cheaper and more pervasive opportunities for combinatorial innovation emerge, making CC is a catalyst for innovation. In addition, while in the traditional business model, 70% of the costs related to ICT is spent on maintenance of infrastructure and applications, these costs may be drastically reduced under the CC model since the client company need not buy, maintain and upgrade hardware or software [21]. Companies opting for the CC model avoid IT problems and can thus focus on their core competencies and innovation.

According to IDC [22], CC begins to fall in the acceleration phase of adaptation and it is expected that up to 25-40% of enterprises will use CC by 2012. BCG analysts [4] estimate that in 2012 the CC market value will reach 60 to 80 billion dollars encompassing 10% of the worldwide ICT market. According to the CEBR report [23], in the years 2010-2015 the benefits of the use of CC in France, Germany, Great Britain, Italy and Spain will reach 763 billion euro - approximately 1.57% of the summed GDP of these countries. It thus makes sense to expect that the macroeconomic impact of CC may be relatively large, especially on GDP growth [24], as was the case with the emergence of ICT and its impact on development of economies. To estimate the impact of CC use on the GDP of individual countries of the Baltic Sea Region, we employed the method discussed in the next section.

3 Method and Data

In this section, we try to estimate the value of production resulting from the use of CC on the GDP of individual countries of the Baltic Sea Region. Estimated data we obtained on the basis of GDP, the percentage of ICT value in GDP creation and the percentage of CC spending in ICT (aggregated for the whole world). This is a simplification, which does not result directly from a traditional Cobb-Douglas production function [25]. In this function, on the effect of production, in addition to capital input, influence the labor input and total factor productivity (TFP). Thus, our results are not exact, but only estimated. Nevertheless, they reflect the fact that the adaptation of the CC may be an important resource that affects the GDP. It should be noted that in the future, additional tests proving their compliance should be conducted. The greatest simplification is the adoption of the aggregate value 4% share of CC in the investment in ICT [26]. It is the aggregate for the whole world economy, not to each of the Baltic Region countries separately. In reality its value may be completely different for each country, the same as it is with ICT share in GDP of individual countries within the comparison, but the right data for CC are not available so far.

All the data that have been used to the calculations, contains Table 1. The data contained therein require an additional comment.

The data on gross domestic product (GDP) at purchasing power parity (PPP) exchange rates is the sum value of all goods and services produced in the country valued at prices prevailing in the United States. This is the measure most economists prefer when compare resources across countries. The data were taken from the report "The World Factbook", available online by the Central Intelligence Agency [27]. GDP height is given in dollars. Please note that the value of GDP and the share of investment in its creation relates to the period just before and at the beginning of the recent crisis, which also spread to the different pace for the individual countries. It also had a different impact on developed countries (e.g. Sweden and Germany) and developing countries (e.g. Lithuania, Latvia and Poland – what's more, even in this group the impact of the crisis on the economy was different).

The information on the ICT share in GDP were taken from the Eurostat statistics [28]. For most of the countries available data are from 2008. Only for Denmark and Sweden provided data are from 2007. We also couldn't find data for Norway and Russia. The data for Latvia are taken from Edvins Karnitis' paper titled "ICT as catalyst for economic development: strengths and weaknesses" [29]. The share of ICT in the GPD varies from 2,19% to 8,5%, but unfortunately the second value is not sure.

Multiplying the GDP value by a percentage of the ICT sector we gained the value of ICT sector. Then multiplying obtained value of ICT sector by approximated worldwide aggregate percentage of the Cloud Computing spending in ICT (4%) we gained the estimated aggregate value of the Cloud Computing. As mentioned earlier, it's particularly risky to assume the same 4% value as a share of expenditures on CC in total spending on ICT. Further studies in this field will be necessary. At this stage we can only hypothesize that the value of GDP, ICT and CC are interrelated - the higher the share of ICT in GDP, the higher share of CC in ICT.

4 Results

Data in the Table 1 clearly divided examined countries in two groups in terms of ICT investment (and hence the CC). The Nordic countries have the highest share of ICT in GDP (for example, Finland has more than doubled share than Poland). It is also a large discrepancy concerning Baltic states: Lithuania, Latvia and Estonia. The share of ICT in Estonia's GDP is more than double that of Lithuania (Latvia almost four times higher, but these are not official data, but estimates).

Table 1. Aggregate value of cloud computing in Baltic Region Countries' in 2008

Country	GDP in 2008 (purchasing power parity)*	Percentage of the ICT sector on GDP (2008)**	Value of ICT sector on GDP	Worldwide aggregate percentage of the Cloud Computing spending in ICT*****	Aggregate value of the Cloud Computing on GDP
Denmark	$210,4 billion	5,24%***	$10,52billion	4%	$0,42 billion
Estonia	$27,96 billion	4,52%	$1,26billion	4%	$0,05 billion
Finland	$197,6 billion	7,06%	$13,95billion	4%	$0,56 billion
Germany	$2 998 billion	4,17%	$125billion	4%	$5 billion
Latvia	$39,99 billion	~8,5%****	$3,39billion	4%	$0,13 billion
Lithuania	$65,72 billion	2,19%	$1,44billion	4%	$0,06 billion
Norway	$276,2 billion	******	******	4%	******
Poland	$687 billion	3,42%	$23,49billion	4%	$0,94 billion
Russia	$2 331 billion	******	******	4%	******
Sweden	$358,4 billion	6,54%***	$23,439billion	4%	$0,49 billion

Source: own estimates based on:

* The world factbook [27]

** Eurostat [28], data for 2008

*** Eurostat [28], data for 2007

**** ICT as catalyst … [29]

***** IT Cloud Services Forecast … [26]

****** no data

5 Summary

As we have shown, although the CC is in the early stages of adaptation and growth, it has a significant impact on the development of economies. Mentioned IDC research [22] shows that CC will be expanding its participation in the ITC. So we can assume that its share in GDP of various countries will increase, and its faster adaptation will increase GDP.

Etro [3] has estimated the impact of CC on the growth, taking into account such factors as the dynamics of entry / exit of companies on the market in various industries and dynamics of employment. He studied the impact of the reduction in the cost of market entry by moving the capital spending (CAPEX) on operational costs (OPEX). However, many questions still remain unexplained and unexplored.

The paper is the starting point and the first step, which begin our research on the impact of cloud computing on the economy and GDP:

1. Innovation drives the economic growth. In our opinion CC has a significant influence on it (because of the low cost market entry). So far, the implementation of ICT solutions required large financial outlays. Their refund were not sure,

therefore they were reserved mainly for large companies. With CC, just idea and relatively low spending (in relation to the traditional model of buying ICT) is enough.

2. CC allows for low cost and quick implementation (and flexible forgoing them) of large test and production environments . This allows testing of innovative solutions to a much less cost (CAPEX -> OPEX). To what extent is speeds up innovation and speeds up entry of new products on the markets? We can forecast how many firms can arise due to CC, but how many of them build their value through the CC? How many inventions / new solutions is the result of the application of the described model? How much of R & D departments benefits from this? What CC contributed to the GDP of different countries?
3. Mentioned report by BCG [4] says on the impact of CC on the creation of new models of cooperation and new business processes within companies. How many previously impossible business ideas has been realized thanks to CC? How they affected the economy?
4. Productivity and efficiency. ICT increases work efficiency. For what part of this increase CC is responsible? How it is changing along with the adaptation of CC model?
5. The problem of convergence. How CC helps developing countries to build their GDP to catch up with the rich?
6. How the amount of income of the country determines the adaptation of CC? What is the difference in the amount of investment and the use of the CC between developed and developing countries? The results of this article do not respond to it directly. Germany which is much more developed country (GDP per capita) spend much less on CC (per capita) than some of the Baltic states with lower GDP per capita.
7. How the level of education affects the adaptation of the CC in different countries?
8. How foreign trade and international competition stimulates adaptation of CC?

References

1. Aghion, P., Howitt, P.: Endogenous Growth Theory. MIT Press, Cambridge (1997)
2. Colciago, A., Rossi, L.: Endogenous Market Structure and Labor Market Dynamics (2011), `http://mpra.ub.uni-muenchen.de/29311/1/MPRA_paper_29311.pdf` (access May 20, 2011)
3. Etro, F.: The Economic Impact of Cloud Computing on Business Creation, Employment and Output in Europe. Review of Business and Economics (2), 179–208 (2009), `https://www.econ.kuleuven.be/tem/jaargangen/2001-2010/2009/2009-2/RBE%202009-2%20-%20The%20Economic%20Impact%20of%20Cloud%20Computing.pdf` (access May 20, 2011)
4. Dean, D., Saleh, T.: Capturing the value of cloud computing. BCG (2009) `http://www.bcg.com/documents/file34246.pdf` (access May 20, 2011)
5. European Commission: ICT and e-Business for an Innovative and Sustainable Economy, 7th Synthesis Report of the Sectoral e-Business Watch (2010), `http://www.empirica.com/themen/ebusiness/documents/EBR09-10.pdf` (access May 20, 2011)

6. Carr, N.: IT Doesn't Matter (2007), http://www.roughtype.com/archives/2007/01/it_doesnt_matte.php (access May 20, 2011)
7. Jorgenson, D.W., Vu, K.: Information Technology and the World Growth Resurgence. German Economic Review 8(2), 125–145 (2007), http://www.economics.harvard.edu/faculty/jorgenson/files/ITandWorldGrowthResurg_GermanEconomicReview.pdf
8. OECD: The economic impact of ICT: evidence and questions (2006), http://ec.europa.eu/information_society/eeurope/i2010/docs/high_level_group/note_on_economic_impact_of_ict.pdf (access May 20, 2011)
9. Vu, K.: Measuring the Impact of ICT Investments on Economic Growth (2005), http://www.hks.harvard.edu/m-rcbg/ptep/khuongvu/Key%20paper.pdf (access May 20, 2011)
10. Nasab, E.H., Aghaei, M.: The Effect of ICT on Economic Growth: Further Evidence. International Bulletin of Business Administration (5), 46–56 (2009), http://www.eurojournals.com/ibba_5_05.pdf (access March 11, 2003)
11. Kobyliński, A.: Cloud Computing – Novelty or The Emperor's New Clothes. In: Education, Research & Business Technology, The Proceedings of the 10th International Conference on Informatics in Economy, Bucharest, Romania (2011)
12. Mell, P., Grance, T.: The NIST Definition of Cloud Computing. National Institute of Standards and Technology (2011), http://csrc.nist.gov/publications/drafts/800-145/Draft-SP-800-145_cloud-definition.pdf (access May 20, 2011)
13. Amazon Elastic Compute Cloud, http://aws.amazon.com/ec2/ (access May 20, 2011)
14. Windows Azure, http://www.microsoft.com/windowsazure/ (access May 20, 2011)
15. Google App Engine, http://code.google.com/intl/pl-PL/appengine/ (access May 20, 2011)
16. Facebook, http://www.facebook.com (access May 20, 2011)
17. Quest Diagnostics, http://www.questdiagnostics.com/ (access May 20, 2011)
18. Google Health, http://www.google.com/intl/pl-PL/health/about/ (access May 20, 2011)
19. Microsoft HealthVault, http://www.healthvault.com/personal/index.aspx (access May 20, 2011)
20. Brynjolfsson, E., Hofmann, P., Jordan, J.: Cloud Computing and Electricity: Beyond the Utility Model. Communications of the ACM 53(5), 32–34 (2010), http://delivery.acm.org/10.1145/1740000/1735234/p32-brynjolfsson.pdf?key1=1735234&key2=6394474031&coll=DL&dl=ACM&ip=89.79.201.193&CFID=19668103&CFTOKEN=66933069 (access May 20, 2011)
21. Cloud Computing: The Next Big Thing? World Economic Forum (January 30, 2009), http://www.weforum.org/sessions/summary/cloud-computing-next-big-thing (access May 20, 2011)
22. Cloud Computing: Preparing for the Next 20 Years of IT. IDC, Singapore (2009), http://www.idc.com.sg/CloudComputing2009/ (access May 11, 2011)

23. The Cloud Dividend: Part One. The economic benefits of cloud computing to business and the wider EMEA economy. France, Germany, Italy, Spain and the UK. CEBR (2010), http://uk.emc.com/collateral/microsites/2010/cloud-dividend/cloud-dividend-report.pdf (access May 20, 2011)
24. OECD Information Technology Outlook, OECD (2010), http://www.oecd.org/document/20/0,3746,en_2649_33757_41892820_1_1_1_1,00.html (access May 20, 2011)
25. Cobb, C.W., Douglas, P.H.: A theory of production. The American Economic Review 18(1), 139–165 (1928), http://www2.dse.unibo.it/golinell/teaching/corsi/corsi_ea/ea_aq/CobbDouglas_AER_1928.pdf (access May 20, 2011)
26. Gens, F.: IT Cloud Services Forecast – 2008, 2012: A Key Driver of New Growth. IDC (2008), http://blogs.idc.com/ie/?p=224 (access May 20, 2011)
27. The world factbook. CIA (2011), https://www.cia.gov/library/publications/the-world-factbook/index.html (access May 20, 2011)
28. Percentage of the ICT sector on GPD. Eurostat (2011), http://appsso.eurostat.ec.europa.eu/nui/show.do?dataset=isoc_ic_bisec&lang=en (access May 20, 2011)
29. Karnitis, E.: ICT as catalyst for economic development: strengths and weaknesses, http://www.irfd.org/events/wf2005/papers/karnitis_edvins.pdf (access May 20, 2011)

The Viable Systems Approach (VSA) for Re-interpreting Network Business Dynamics

Paolo Piciocchi[1], Clara Bassano[2], Erica Paduano[3], and Maureen Galvin[4]

[1] Department of Political, Social and Communication Science, University of Salerno, Via Ponte don Melillo, Fisciano, Italy
p.piciocchi@unisa.it
[2] Business Studies Department, "Parthenope" University of Naples, Via Medina, 40, Naples, Italy
clara.bassano@uniparthenope.it
[3] Department of Political, Social and Communication Science, University of Salerno, Via Ponte don Melillo, Fisciano, Italy
eripad@interfree.it
[4] Business Studies Department, University of Salerno, Via Ponte don Melillo, Fisciano, Italy
mgalvin@unisa.it

Abstract. The variety of network-district businesses has in strategic terms, resulted in less efficiency within local systems and reflected negatively on the competitiveness of district production. The paper seeks to re-interpret the dynamics of network-districts from the Viable Systems Approach (VSA) perspective. In such a scenario, the limited or non-use of information and communication technology (ICT) tools risks affecting competitive capacity both in an individual and network-district perspective. The paper highlights how the creativity of the Italian network-district system gains momentum i.e. greater competitiveness by using ICT tools in a systems approach to facilitate cultural and management change and to integrate communication within and outside the network-district system.

Keywords: network-district, dynamics, viable systems approach (VSA).

1 Introduction

Firms can be defined as networks of interdependent phenomena with clear cut or fleeting, short-lived goals. However, understanding business dynamics implies a focus on widespread relational networks and the *systems approach* is an ideal platform to start from. In terms of network-district business organizations, the approach highlights how firms basically reflect the postulates of business models and theories. In particular, as concerns small and medium size enterprises (SMEs), an analysis of Italian industrial districts, indicates a wide gap in cultural/managerial terms accompanied by low propensity for technological innovation [1].

The structural limits of small firms, physical proximity and cultural bias also emerge in the district scenario[2].

Our approach pursues cultural and management change by means of experimental collaborative strategies of integrated communication both in and outside the district

J. Grabis and M. Kirikova (Eds.): BIR 2011, LNBIP 90, pp. 304–320, 2011.

system. If on the one hand, interesting developments in technical and organizational terms are evident, on the other, scarce attention is addressed to investing in information and communication technology (ICT) for communicational efficiency and effectiveness. Furthermore, the structural characteristics of SMEs and relational processes heavily limit the understanding and implementation of ICT.

Our study proposes the conceptual framework of the Viable Systems Approach (VSA) as an interpretative theoretical framework for analyzing network-district dynamics relative to information and communication technology in planning marketing and communication district network strategies for competitive local systems on global markets [3].

If it is true that Business Management studies are facing problems from the perspective of redefining the logics of district dynamics, then the identification of a new interpretative key – appraising the dynamics of industrial districts through the VSA conceptual framework [4], [5], [6], [7], [8] as a model for observing and interpreting complex businesses and social phenomena through interpretative schemes – is extremely relevant.

In this context, two traditional conceptual categories, ‘district’ and ‘network’ examined from a VSA perspective, offer new insights for the analysis of “agglomerations of enterprises” located in areas linked by relational content and characterized by a systems deriving status.

In theory, industrial districts are identified by a series of distinctive features relative to each territorial-productive area [9]:

- large number of enterprises (categories of industrial enterprises);
- marked division of inter-firm labor;
- propensity for specialized production;
- enterprise relational networks in a local community;
- common attitude towards innovation;
- a link between firms in the district and their respective target market.

Although various meanings have been attributed to “industrial district”, Becattini, conceives that: *the district is a socio-territorial entity characterized by the active coexistence, in a circumscribed territorial-area, naturally and historically determined, of a community of people and a population of industrial enterprises* [10]. The community and enterprise incorporate a system of values in common [built and consolidated over time] and a system of institutions and rules [for the transmission of these values from one generation to another].

Districts are therefore, characterized by external economies, generated both by natural factors related to the location of the production community and by intrinsic factors such as a common vocation or “industrial atmosphere”. The latter permeates economic and other relationships and is one of the main drivers of distributed, cognitive and non-cognitive learning processes (learning by doing, by using, by localizing, by specializing, by external adapting and by inter-firm relationships) distinctive and inimitable knowhow relative to a specific industrial district. Typically, economic benefits – i.e. reduction of production costs etc. – encourage the regeneration of firms belonging to the same systems area or located nearby and promote further growth through synergies, thus increasing the economic potential and the cognitive heritage of the whole district: *everyone benefits from the ideas of its neighbors* (…)

and each successful invention, a new car, a new procedure or a new way of organizing activity, is likely to improve once launched. Both large and small businesses can benefit (...) *but these are more important for small businesses* [11]. As Becattini states: (...) *what holds together firms that are part of a Marshall type industrial district* (...) *is a complex and intricate network of external economies and diseconomies of conjunctions and connection costs, a cultural and historical heritage that surrounds both the inter-relationships and those more purely interpersonal* (…) [12].

In the relational structure of territorial entities, in many cases quite distant from the concept of the "Marshall type district" [13], the shift from a global macro-systems view to a natural business micro-systemic view defines the role of small, medium and large enterprises within a geographically limited, highly systemic context or environment.

From a more traditional business studies viewpoint, our paper integrates the analysis of districts by interpreting the structural setting from a systems perspective. The assumption is that as an industrial system is the result not the sum of diversified entrepreneurial activities, it follows that district dynamics and territorial systems cannot be defined without a study of systems *tout court.*

2 Network-District: A Complex Concept in a Relational Key

A brief review of the literature highlights three key elements of inter-personal and inter-intra-district relationships [14], [15], [16]:

- network or system of relationships between internal and external businesses for knowledge and economic objectives;
- stable formal relationships created from informal social ties;
- the function and the importance of nodes for the network macro-system.

Qualifying strategic and organizational business models in which traditional criteria relative to entrepreneurial projects – optimization of resources, effectiveness of results and schematization of objectives, lifelong learning, critical functions – are rethought on the basis of bottom up competitive logics oriented towards cooperation [17] and common objectives, rather than top down control and individualism.

Furthermore, the importance of exploiting inter and intra-firm entrepreneurial skills, is a strategic element for the development of enterprise potential, creating favorable conditions for governing synergies between individual and independent organizations. In the literature, criteria for the effective working of the network system conceived in terms of collaboration between the links/nodes and the coordination of specific functions is highlighted; less emphasis however, is given to the natural process of network formation: the transition from a series of typically social, spontaneous relations shared over time maturing into strong ties and optimizing the performance of the business system as a whole.

Another approach describing the various forms of aggregation, and defining the developmental stages of the process, would highlight the value of interpersonal networks that regardless of the model, are the result of social and economic relationships between individuals belonging to a certain geographical area or community harnessing their knowhow.

The theoretical basis, a conceptual framework for our empirical study, analyses the development of personal networks and entrepreneurial potential, behavior and the critical functions underpinning management of complex social and economic value systems.

From a sociological perspective, relational analyses evidence interesting empirical findings and models easily adaptable to business-economic contexts; networks are represented through recurring interpersonal exchange patterns. This implies that the interpretation of entrepreneur relations is determined by willingness to join collaborative schemes but hindered by limited knowledge [18] of the social, economic and entrepreneurial reference.

In addition, the structure of the network depends on the location of each firm, relational reciprocity and frequency [19]. Notwithstanding, uniformity is fundamental between business entities, i.e. number, trade potential and structure of the ties.

Anthropological and sociological theories and analytical models have given insights into the dynamics of business networks. In particular, relational exchanges can be classified in four categories [20]:

1. Permanent links or ritual, typical of clans where the logic of reciprocal gifts prevails;
2. Links at two levels of trust between individuals from which utilitarian relationships stem;
3. Potlatch links or competitive exchanges establishing orders of prevalence or power, based on the potential exchange value;
4. Utilitarian links, the logic of economic exchange based on trading.

These categories imply that cohesion in business networks logic is utilitarian, based on regulated economic behavior (laws or formal relationships). However, cooperation starts with a shared social structure, defined by customs or informal relationships) [20]. The utilitarian approach in networks should therefore take into account that enterprises are oriented towards collaboration to reach more qualified strategic positioning. This appears evident in district areas, where observation and the emulative action of the enterprises, by differentiating process and/or product, determine conditions for growth proportionally to available resources benefiting all parties concerned.

The district can also be envisaged as a particular type of cluster, characterized by firms associating by chance and reaping unintentional benefits from spatial and sector proximity: deliberate joint action. Collective efficiency characterizes firms that operate within a system, either as a cluster, district and/or a network for competitive advantage, not achievable individually. Moreover, industrial district interaction contemplates the ability to incorporate input from the system and the general environment [20] – social, political and economic – gaining advantage through diversified strategic action.

The appeal of a network perspective for the analysis of a district scenario and its dynamics – the emergence and evolution – of interdependencies between the actors, pivots on the patrimony of experience and knowledge, even external to the local system, at the core of the consolidation process between networks of social and economic of relationships [21], [22].

It should be emphasized that the concept of network of businesses or business network is clearly of Italian origin; in fact, it is a consolidated view that small firms, especially local ones, gain *volente or nolente,* undeniable competitive advantage but also greater opportunities for survival by belonging to a network [23].

The network concept is an interpretative key for clarifying the functioning of district activity above all in terms of devising a joint scheme relative to internal and external relationships within the local system [24].

The survival of small firms for instance, depends on external networks. Power spread, shared, or acknowledged by virtue of interactive mechanisms is therefore, fundamental.

The relationships and networks identified in our analysis lead to the description of district dynamics from a strategic and organizational viewpoint. Empirical findings are illustrated in a diagram representing the nature of relationships and strategic elements upon which network-district success depends.

Fig. 1 and 2 illustrate prevailing collaborative forms that evolve into more regular structures on the basis of social and economic components in the area: networks of social businesses and business networks of an economic nature.

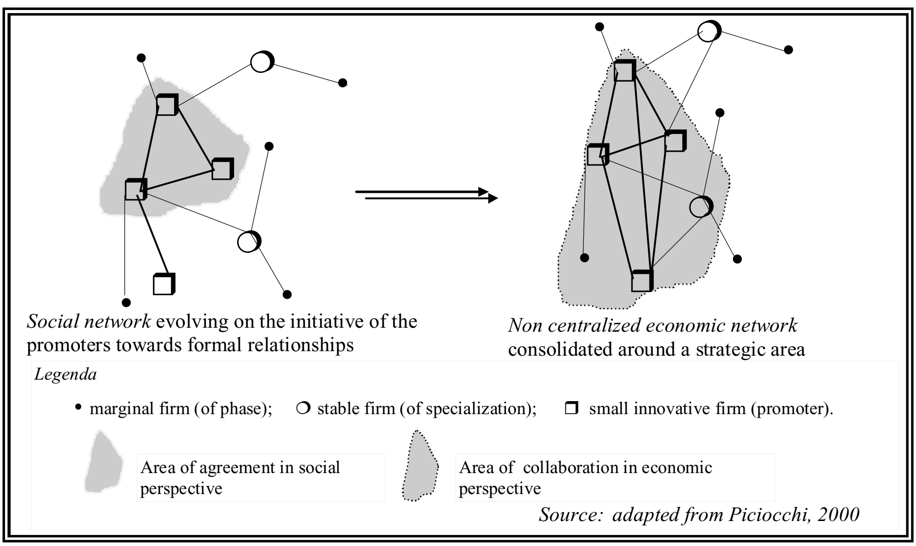
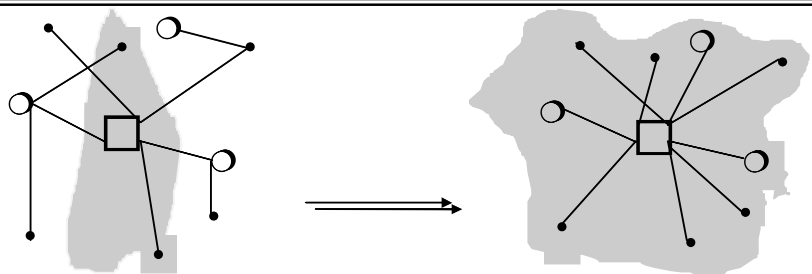


Fig. 1. The evolution of district relations in non-centralized networks

In Fig. 1 a scheme of a decentralized and shared network in management activities (consortia and cooperatives) is illustrated where the strategic actors in the districts are the small innovative enterprises. They rely on mutual social relationships and the propensity for organizational flexibility. Complementary skills and assets tend to be prevalent in the individual phase.



Economic network matrix of relations of dependence of the smaller enterprises compared to the enterprise drivers

Network cantered on enterprise drivers. It extends to block further partner relations outside the network

Legenda

• marginal firm (of phase); ❍ stable firm (of specialization); ☐ medium/great innovative (focal firm).

strategic area of collaboration managed by the focal firm

Source: adapted from Piciocchi, 2000

Fig. 2. The evolution of district relations through network

In Fig. 2 the process of formalizing local relationships by the focal firm (driver) in the district (medium/large and innovative companies) is illustrated. Around the focal firm an area of collaboration is identified in the expanded productive system, the direction and management of which depend on decisions made by the focal firm.

External growth is based structurally on smaller firms as concerns the development phase and work processes (marginal firms) as well as specializations (stable firms). The focal firm [25] having surpassed the phase of internal growth, reduces outsourcing; relations with stakeholders in the district then need to be stabilized to keep activities under control. The regulation of network relationships proceeds through the acquisition of smaller enterprises in crisis [26] exclusive supply relationships and direct investments that consist also in the purchase of machinery and equipment in use.

Consequently, the district may be described as an environment in which entities – viable systems in the VSA perspective – survive in a network configuration characterized by distinctive synergic specificity, i.e. representing the components of a strong inter-systems link (consonance).

3 Network Dynamics from a Systems Perspective: The Viable Systems Approach (VSA) Conceptual Framework

The concept of system and the qualification of firms as a system is not new and can be traced to Beer's theory (1972) [27] however, the recent canonization of the Viable Systems Approach (VSA) [4], [5], [6], [7], [8] has contributed to giving valuable insights into understanding decision-making processes and strategic, operational dynamics – in a word the survival – of business organizations.

In particular, Golinelli stretches the concept and theorizes that firms as viable systems are effectively characterized by:

– structure and system;
– government and operative structure;
– competitiveness and consonance for firm survival.

The concepts of consonance and resonance are clearer if we connect them to the dualism of Structure and System, Government and Operative Structure (the evolution of Beer's theory), the Conceptual Framework, Relation and Interaction.

The *structure-system* dualism clarifies:

– the static and dynamic aspects of organizations (relationships and interactions; level of openness and capability; negotiation structures and processes);
– the concept of complexity which is typical of socio-economic systems (i.e. complexity originates from the differences between the system and the structure on emergence and in relation with the environment).

Particularly, the distinction *structure-system* focuses on the nature of organizations - social and otherwise - undeniably characterized by a logico-physical order (structure) and action addressed to a specific aim (system) [8], [28].

More specifically,

> If we can qualify *structure* as:
>
> *a series of related components, to which a role is assigned in firm processes from which the system emerges*
>
> then the *system* is:
>
> *the structure in act i.e., working towards the achievement of a common goal*

In other words, for each entity it is always possible to identify components that related to specific organizational schemes devised by the government of the system, are activated for survival of the said system through the attainment of strategic objectives. That is to say, given a structural configuration – specific structure – several systems can derive by virtue of the different objectives that such a structure pursues.

In conceptual terms, therefore, the *structure* represents the inter-related logico-physical components that – because of the links in terms of role and the rules of behavior established by the firm's governing body (government) – are activated synergistically in order to pursue the objectives of firm survival. On the contrary, the *system* represents the structure in momentum, or the viable expression of the complex organization at a given moment and in a particular context of reference. A system, therefore, is defined viable if it is able to survive in a specific context, setting the foundations for structural compatibility – consonance – to the extent that it ensures an adequate level of connectivity, or the definition of a common system of values addressed to the pursuit of shared evolutionary paths – resonance – [4].

When the system emerges from the structure, *relations* and *interactions* prevail over structural components and consequently the firm as a system begins to evolve in an increasingly complex context.

In order to survive (survival is seen as the final target of any viable system) the firm seeks:

– competitiveness, – the achievement of a cost and/or quality advantage;
– consonance, – consensus building within the firm and with its supra-systems (i.e. systems populating the firm's environment and sharing project expectations and pressures).

In this regard, the concept of *systems identity* is pertinent.

Viability, in effect, represents the ability of an enterprise to survive in a specific context through processes of adaptation, constantly seeking consonance and resonance with other systems for the constant exchange of resources. Consequently, systems identity in its general meaning, refers not only to the system's ability to represent itself adequately (by means of planned communication forms) but also that of being perceived by its targets as an entity capable of satisfying stakeholder expectations (by means of spontaneous and/or not strictly planned communication forms). In other words, how information is elaborated by the government and communicated to the operative structure for the implementation of strategic plans is decisive for ensuring systems viability. In this sense, ICT definitely contributes to making network structures more flexible and adaptable to stakeholder needs thanks to more efficacious and streamlined communication and interaction between the entities of the network [29].

In this perspective, smart information logics based on *Information Logistics* alters the nature of competition accelerating the flow of information and that of new products and services. In particular, the potential of information logistics spread across the strategic and operational nodes of a network enables businesses to organize themselves differently, to provide and distribute new goods and services: firms have to face the fact that competitive advantage can appear and disappear overnight, but if shared and distributed, it is likely to be more lasting. Consequently, *Information Logistics* and networks contribute synergistically to enhancing systems; networks are sources of competitive advantage creating value centers both inside and outside the firm.

It goes without saying that the inside value chain (in-sourcing) is no longer an adequate model for representing the mutual dependence and "multi-nodes" created between viable firms/systems leading to outsourcing, risk sharing and efficiency factors. Thus, a suitable model is the shared value creation process, a "spider web" of often independent collaborative skills and expertise. Smart network systems of information guarantee the necessary collaborative variety for researching, creating, processing and distributing value as required by the market. This means that it is not just a simple, generic process of sharing material resources, but rather the identification of processes and algorithms that describe how to act in specific scenarios, making use of information relative to the product itself. In conclusion, the benefits of applying the *Information Logistics* approach favors the creation of processes and network systems focused on the benefit of the market, increasing productivity, information transparency and the rapid adaptation of processes.

In terms of *systems identity* (isotropy in VSA terms) two other conceptual categories require clarification: government and the operative structure. Government comprises the decision making area, the top decision making process and entity – individual and/or group – in any viable system. The operative structure is the action area; its function, by means of self-organising processes, being to carry out the decisions made by government.

Each viable system respects isotropy in the sense that, beyond the possible structural configurations (forms) and the systemic actions (demonstrations) an area of decision making (government) and an area of action (operative structure) are always recognizable.

VSA is based on a methodological framework in which the vision of the firm as a viable system is firmly grounded in a series of postulates and its dynamics framed within a conceptual matrix that defines the cycle; the latter, starting with the business idea, ends with the firm as a system [4], [5]. This methodological pathway – midway between the traditional analytical approach (focus on the parts) and the holistic approach (focus on the whole) based on relations and interactions – refers to "postulates" that can be summed up as follows:

Postulate 1: A system is viable if it can survive in a specific environment
A viable system enjoys a certain degree of autonomy but is contextually associated in an environment justifying its presence and its function (ongoing processes of adaptation)

Postulate 2: Viable systems and isotropy
Viable Systems have the same identity characterised by the interacting co-existence of two distinct areas: a decision making area (government) and an action area (operative structure)

Postulate 3: The viable system in the pursuit of purposes and objectives is linked to supra-systems and subsystems from which and to which, expectations, guidelines and rules can be received and allocated
Firm survival depends on the capability and suitability of viable systems (firm systems) to satisfy the expectations of the supra-systems and address the goals and objectives of sub-systems

Postulate 4: A viable system, as an autonomous entity, is merged within the supra-system of reference in a specific time-frame by virtue of processes of resonance which may follow conditions of consonance
Relations between system and supra-system imply structural compatibility in the exchange (consonance). Compatibility produces harmony of purpose within the supra-system (resonance)

In business management, the viability of a system depends on the ability of the government to develop conditions of consonance and resonance with the relevant entities in the context (supra-systems).

Not all the entities are fully accomplished viable systems however; the degree of fulfillment depends exclusively on the presence within the structure of government – management or ownership system – which drives the system in its survival process. In this sense, the VSA conceptual framework suggests three hypotheses of "systems" entities:

– In Embryo Systems (markets);

- Evolving Systems (networks, districts);
- Viable Systems.

In embryo systems are organizations in which clear traces of an individual or Board to govern the system is lacking. Inter-component relationships are activated on the basis of market transactional logics. In evolving systems, assimilated within the network configurations – centralized or otherwise – there is no effective configuration of government. However, decision making centers reporting both for a single entity (focal firm) and centers (clusters of enterprises or defined groups of enterprises) are evident. Here, the ability "to trace the evolutionary paths" of the whole system clearly emerges: government consolidates, often informally, its own decisional role within the network [30].

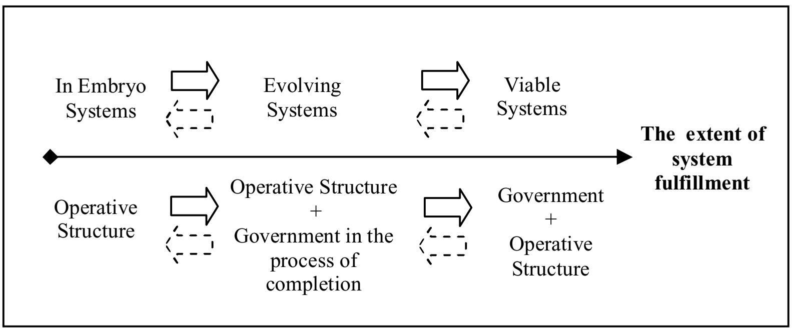


Fig. 3. The continuum evolving systems to viable systems. Source: Piciocchi et Bassano, 2009 adapted from Liguori et Iannuzzi, 2008

Viable systems refer to the full correspondence of system isotropy (systems identity) where the clear and explicit activity of government relative to a specific structure is evident.

Given the nature of the industrial district networks and considering their generative and evolutionary path, in a VSA perspective, a network is interpreted as the configuration of evolving systems. In fact, systems identity is achieved through government to which implicitly or expressly, the role of guidance and "entrepreneurial example" is acknowledged by all organizations present in the relevant territory. Normally, such coordination is achieved through an informal bottom-up process or responding as a preliminary step, to the related market and eventually, to relationships (networks) if the relationships persist over time and finally, in theory, evolving toward a viable system whereby structural relationships and systems interactions are acknowledged as stable and recurrent [31].

The coordination of the relationships within a local network or the opportunity of transforming a district-system into a viable system imply the ability of government to guarantee the consolidating of internal and external links with the territorial structure on the basis of specific criteria [32], [30]:

- coordination of internal, external, strategic and operational activities of the enterprise and common purpose of the local system;

- motivation of firm members creatively and decisively participating in the joint management of activities and knowledge sharing in the local network;
- the temporary nature or the flexibility of the network configuration, by virtue of the variability of structure and system, subject both to continuous redefinitions on the basis of experience accumulated and of reciprocal and environmental input;
- the productivity of the enterprise and general system in terms of specific contributions of each component to general aims;
- the reliability on the abilities and on the strategic and operational contributions of each component for the general purposes of the network system.

In this respect, Fig. 4 illustrates the scheme of the evolutionary cycle of districts (CED), from genesis to growth, consolidation and/or mortality of the district system [33].

The model of the Viable Cycle of District Systems constitutes therefore, a descriptive hypothesis of the evolutionary phases of inter-personal relationships between economic entities – or otherwise – in the district, as well as a means of analysis for the planning and the modification of relationships, to achieve viable district systems, generating value for the components (nodes) and for the general system as a whole.

Relational dynamics develop on the basis of the recognition of a need for collaboration and the identification of suitable partners with whom to pursue such aims, primarily social and subsequently, economic. The conceptual framework of the VSA provides an opportunity of studying relationships in terms of the dynamics of entrepreneurial action, gradually shifting from the generative phase (expectations, communications, actions and reactions, appointments, problematic situations and conflicts, spontaneous and/or planned activities) to that of the relational system itself.

The dynamic pathway described in the Viable Cycle of the District Systems model, in terms of implicitly or explicitly interactive local components, implies simple market transactions and the creation of inter-personal networks (in embryo systems or system areas), economic and informal social networks (evolving systems or districts) that only in the case of joint and acknowledged government can evolve into formal networks (viable systems). Particularly in the in embryo phase, potential partners in the face of uncertainty (of the relationship) spontaneously and naturally, favor collaboration (still latent) or join ranks with one or more partners functional to their activity. Sustaining this informative exchange is the search for mutual value from the formalization of the relationship in terms of experiences, competence and additional knowledge created by interaction [32].

Experience in the initial phase and elements of strategic and operational agreement help to reduce the uncertainty of collaboration, the extent to which the local components are involved and their relational value; not to mention further policies of mutual satisfaction. Stable relationships underline the potential value of networks by virtue of linear action and communication within the district. During the phases of introduction and growth/development, organizations in interactive terms, are characterized prevalently by informality, elevated instability and the uncertainty of future results (in embryo system vs. evolving system).

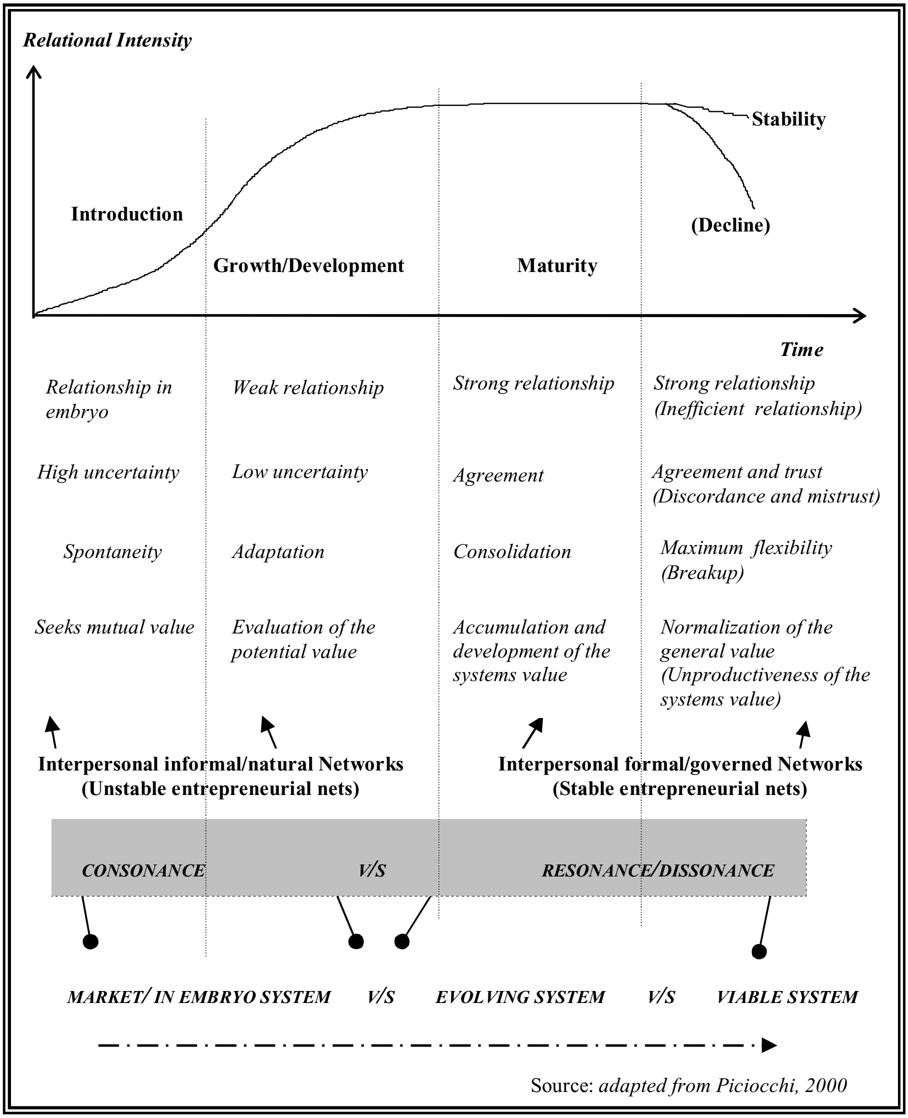


Fig. 4. The Viable Life Cycle of District Systems

When evolving relationships become firmly rooted (phase of maturity) interaction is consolidated in formal bonds. The intensity of the relationship increases the value of exchange enabling further growth, better collaborative schemes and general potential of the territorial system. Therefore, organizational and operational strategies positively influence the relational intensity of the network providing a stable configuration for optimizing node activity and global performance.

As illustrated in Fig. 4, transition to the stability phase brings the network to its final stage of growth. Stability, agreement (resonance) in relationships lead to fiduciary exchange, where the stable value of the relationships or the consolidation of the local system are pursued. Shared experience, expectations, flexibility and adaptations reach elevated levels of consonance and resonance and this decrees in full the formalization of the network system (evolving system v/s viable system), on the basis of a complex relationship that targets economic-financial, social and psychological benefits. The strong interdependence typical of the phases of maturity and stability places the entrepreneurial actors within a formal network, in which activities are governed through stable and encoded functional regulations.

The Viable Cycle of District Systems model represents two types of networks-districts:

- *natural or informal districts,* in the Introduction and Growth phase;
- *formal or governed districts,* in the Maturity and Stability phase.

4 Conclusions

Network-district competitiveness has to be analyzed from both a micro and macro perspective where:

- micro refers to each firm;
- macro refers to the entire district system.

However, both are penalized by opportunistic relational processes existing between the many stakeholders of the network-district and this does not facilitate external competitiveness [34].

In this context, ICT by virtue of its underlying capacity for creating networks, guarantees a platform from which to launch a permanent learning process able to convey the image, reputation and identity of the entire local system to the global market [35], [36].

As is well known, small district firms – in particular on the Italian scenario – have to date, invested in innovation merely for promoting production [37]. The resultant gap can only be narrowed by informed awareness of the new ways of communicating – cultural management change – and the will to experiment collaborative strategies of communicational integration within the network-district system [38].

In our paper, the appeal of the conceptual category "network" and the VSA conceptual framework clarify the systems dynamics of districts, their structural characteristics and the roles and synergies of each organized component. In our approach the analysis of relationships within enterprises united by strong productive specialization and systems vocation is clearly delineated.

The attraction of approaches such as VSA and Networking is justified in terms of interpreting "productive districts", not as closed circuit systems, but as networks of structural relationships capable of producing consonance and resonance within and outside the district's geographical boundaries. In fact, in a network and systems perspective, enterprises are not limited by interaction imposed by a system of values but rather are able to weave a web of adequate relationships to guarantee both the exchange of critical resources for business survival and at the same time, to consolidate the territorial vocation of the local system.

Furthermore, the concept of district, understood as cluster of enterprises capable of profiting from external economies and from spatial and sector proximity, tends to disappear with the emerging of an entity of government (focal firm) as a viable centre of the district and the evolving of local systems toward a viable system. Complex systems – such as districts – are defined as systems in evolution towards a viable system if collaboration in and the coordination of the various sub-systems (components of the network) are governed through a spontaneous bottom-up type process or systems entity (for instance the entrepreneurial association of the district) that not only mediates between structural components but above all, plans and implements survival in compliance with VSA postulates and precepts. In other words, the concept of evolving district, envisages synergic integration – of consonance and resonance initially spontaneous and then gradually planned – by virtue of productive specialization and the territorial integration of mainly small size enterprises. Such districts, envisage levels of coordination and collaboration, flexible adaptation and/or transformations driven by aggregation that the government of disseminating information (scientific, technical-productive, commercial etc) circuits makes possible to increase competitiveness, both on a local sub-system (firm) and global level of system (district). The building and maintaining of conditions of structural consonance (relationships) and of systems resonance (interactions and flows) require in fact, non-negligible costs of coordination and control; consequently, the role of each local component is enhanced within the survival logics of the general system.

A further consideration is the nature of the exchanges between the network components; when districts are created and evolve, it is possible to distinguish two categories of relationships: direct and indirect. The direct relationship concerns the intensity – the number of links in the system – developed between the territorial components or the entrepreneurial organizations that adhere to the project of collaboration; i.e. the intensity of the relationship and the effective value of the synergies of the network. The indirect relationship, instead, refers to extensive network interaction, determined by the fact that each component, albeit participating autonomously and freely in the life of the system, implicitly brings a patrimony of previous relations with external organizations; these indirect exchanges affect network operations and increase the intensity of the relationship and its value. Both the direct-formal relationship and the indirect-informal relationship characterize the intensity and therefore, the nature of interactions in the district system.

In sum, the above considerations, albeit partial, confirm that when the characteristics of a district such as the environment-market tend to change (to expand its economy) by pursuing a driver entity (Focal Firm) policy, the district evolves from a system in embryo to a viable system whereby ownership (isotropy or systems identity) decision making (government) and action (operative structure) are able to ensure systems survival.

In ICT terms, the *systems identity* of the district is ensured by intranet networks while the community of district relations, impose the use of interactive marketing tools [38]. Consequently, a *change of perspective* in web communication activities is also needed. This requires the capacity of enterprise to create opportunities for dialogue with the stakeholders on the network by exploiting ICT strategies and tools. In this context, at the present time, the necessary tools for such interaction and dialogue are rarely present in networks-districts and in the event they are available they are not adequately developed.

In conclusion, the practical implications of this scenario demand ongoing and developing intra-inter district learning processes through information and communication technology, to promote competitiveness and growth not only of networks-districts, but also of the individual enterprises that make them up.

Consequently, despite its limits, our paper suggests interesting implications for future research relative to the debate on the use of information and communication technology in the context of marketing and communication strategies in network-districts.

References

1. UNIDO: Development of clusters and networks of SMEs. The UNIDO programme (2001), http://www.unido.org/doc/331111.htmls
2. Granovetter, M.: Economic Action and Social Structure: The Problem of Embeddedness. American Journal of Sociology 91, 481–510 (1985)
3. Arora, A., Fosfuri, A., Gambardella, A.: Markets for Tecnology: The Economics of Innovation and Corporate Strategy. MIT Press, Boston (2001)
4. Golinelli, G.M.: L approccio sistemico al governo dell'impresa, Cedam, Padova (2000)
5. Golinelli, G.M.: Viable Systems Approach (VSA). Governing Business Dynamic, Cedam, Kluwer (2010)
6. Golinelli, G.M., Spohrer, J., Barile, S., Bassano, C.: The evolving dynamics of service co-creation in a viable systems perspective. In: The 13th Toulon-Verona Conference Proceedings of the International Conference in Coimbra, Portugal, September 2-4 (2010)
7. Golinelli, G.M.: L Approccio Sistemico Vitale (ASV) al governo dell'impresa. Verso la scientificazione dell'azione di governo, Cedam, Padova (2011)
8. Barile, S.: Contributi sul pensiero sistemico in economia d'impresa, Centro Studi di Tecniche Aziendali – ARNIA, n. 18, WM Stampa Editoriale s.r.l., Atripalda, AV (2000)
9. Becattini, G.: Il distretto marshalliano come concetto socio-economico. Studi e Informazione Quaderni (34) (1991)
10. Becattini, G.: Modelli locali di sviluppo, Il Mulino, Bologna (1989)
11. Varaldo, R., Ferrucci, L. (eds.): Il distretto industriale tra logiche di impresa e logiche di sistema, Franco Angeli, Milano (1997)
12. Becattini, G.: (a cura di) Mercato e forze locali: il distretto industriale, Il Mulino (1987)
13. Marshall, A.: Principles of Economy, Utet (1972)
14. Schon, D.A.: Champion for Radical New Invention. Harward Business Review (2) (1963)
15. Maidique, M.A.: Entrepreneurs, champions and technological innovation. In: Hax A.C, To the Guide of the Future, Ipsoa (1989)
16. Hauschildt, J., Chakrabarti, A.K.: Arbeitsteilung im Innovationmanagement. In: Zeitschrift für Organisation (June 1988)
17. You, J., Wilkinson, F.: Competition and Cooperation: Towards Understanding Industrial Districts. Review of Political Economy 6, 259–278 (1994)
18. March, J.G., Simon, H.A.: Organizations. Wiley, New York (1958)
19. Staber, U.: The Structure of Networks in Industrial Districts. International Journal of Urban and Regional Research 25, 537–552 (2001a)
20. Collins, R.: Teorie sociologiche, Il Mulino, Bologna (1988)
21. Camagni, R.: Piano strategico, capitale relazionale e community governance. In: Pugliese, T., Spaziante A (a cura di) Pianificazione strategica per le città: riflessioni e pratiche, Franco Angeli, Milano (2003)

22. Becattini, G.: La campagna di ricerche sui distretti industriali. In: Bellandi, Russo (a cura di), Distretti industriali e cambiamento economico locale, Torino, Rosemberg & Sellier (1994)
23. Lorenzoni, G., Boari C., Grandi A.: Le organizzazioni a rete: tre concetti di base. In: Lorenzoni, G. (eds.), Accordi, reti e vantaggio competitivo. Le innovazioni nell economia di impresa e negli assetti organizzativi, Etas Libri, Milano (1992)
24. Piciocchi, P.: Sistemi di reti imprenditoriali tra le PMI meridionali: relazioni tra imprese nel distretto e nella filiera dell Agro Nocerino Sarnese, Monografia n. 15 della Collana di Studi Aziendali e di Marketing diretta dal prof. Roberto Aguiari – Centro Studi di Tecniche Aziendali ARNIA, WM Stampa Editoriale s.r.l., Atripalda, AV (2000)
25. Bassano, C.: Dalla Crisi dell'impresa-guida alla crisi della rete di impresa: verso un modello di analisi, with Gianluca Vagnani. Esperienze d'Impresa, Boccia Editori, Salerno (2) (2003)
26. Piciocchi, P.: Crisi d impresa e monitoraggio di vitalità. L approccio sistemico vitale per l analisi dei processi di crisi. Collana del Dipartimento di Studi e Ricerche Aziendali – Università degli Studi di Salerno, Giappichelli (17) (2003)
27. Beer, S.: Brain of the firm. The managerial cybernetics of organization. The Penguin Press, London (1972)
28. Siano, A., Vollero, A., Siglioccolo, M.: The Governance-Structure-System Model: a Framework for Corporate Communication Management. Empirical Evidences from some Case Studies. In: Corporate and Marketing Communications Conference, London, Middlesex University Business School, April 16-17, pp. 16–17 (2007)
29. Asheim, B., Cooke, P.: Local learning and interactive innovation networks in a global economy. In: Malecki, J., Oinas, P. (eds.) Making Connections: Technological Learning and Regional Economic Change. Ashgate, Aldershot (1999)
30. Piciocchi, P., Bassano, C.: Governance and viability of franchising networks from a Viable Systems Approach (VSA). In: Giannini (ed.) The 2009 Naples Forum on Service. Service Dominant Logic, Service Science and Network Theory, Proceedings of the International Conference in Capri, Napoli, June 16-19 (2009) ISBN13: 978-88-7431-452-2
31. Amin, A., Cohendet, P.: Organisational learning and governance through embedded practices. Paper Presented at Workshop Information Processes and Path-dependent Evolution: Local Systems Response to Changes in Context, Padua University, November 27 (1999)
32. Becattini, G., Rullani, E.: Local systems and global connections: the role of knowledge. In: Cossentino, F., Pyke, F., Sengenberger, W. (eds.) Local and Regional Response to Global Pressure. The Case of Italy and its Industrial Districts, International Institute for Labour Studies, Geneva (1996)
33. Bassano, C.: I distretti orafi italiani, ESI, Napoli (2008)
34. Calvelli, A., Cannavale, C., Canestrino, R.: The replication of the Italian Industrial district model: cross-cultural issues. Journal of Cross-Cultural Competence and Management (4), 39–74 (2005)
35. Piciocchi, P., Basile, G., Vollero, A., Bassano, C.: Communication through internet-based Technologies (IBT) for promoting competitiveness in Italian jewelry districts. In: Proceeding Act 14th International Conference on Corporate and Marketing Communications hosted by: Faculty of Marketing Department, Business School, University of Nicosia, Cyprus, April 23-24 (2009) ISBN: 978-9963-634-59-0
36. Wenger, E.: Communities of Practices: Learning, Meaning, Identity. Cambridge University Press, Cambridge (1998)

37. Piciocchi, P., Saviano, M., Bassano, C.: Il controllo di gestione nelle reti di franchising. In: Antonelli, V., D Alessio, R. (eds.) Casi di Controllo di Gestione, Ipsoa (2004)
38. Piciocchi, P., Saviano, M., Bassano, C.: Network Creativity to Reduce Strategic Ambiguity in Turbulent Environments: a Viable Systems Approach (VSA). In: The 11th International Conference on Global Business & Economic Development, Proceedings of the International Conference at the University of Bratislava, Slovak Republic, May 27-30, vol. 3 (2009) ISBN – 13: 978-0-9797659-8-8; ISBN-10:0-9797659-8-6
39. Lorenzoni, G.: Imprese, relazioni fra imprese, distretti industriali nello sviluppo delle PMI. In: Varaldo, R., Ferrucci, L. (eds.) Il distretto industriale tra logiche di impresa e logiche di sistema, Franco Angeli, Milano (1997)
40. Liguori, M., Iannuzzi, E.: Un modello di analisi per la definizione del grado di compimento dei sistemi vitali. In: Barile, S. (ed.) L impresa Come Sistema, Giappichelli, Torino (2008)
41. Emery, F.E. (ed.): La teoria dei sistemi. Presupposti, caratteristiche e sviluppi del pensiero sistemico. F. Angeli, Milano (1974)
42. Piciocchi, P., Bassano, C., Paduano, E., Papasolomou, I.: The Virtuous circle of Syntropy (VCS). An interpretative chaos vs cosmos model for managing complexity. In: The 2nd Annual Euromed Conference, Proceedings of the International Conference at the University of Salerno, Fisciano, Italy, October 26-28 (2009), doi:10.3292
43. Rullani, E.: The Industrial District (ID) as a cognitive system. In: Belussi, F., Gottardi, G., Rullani, E. (eds.) The Technological Evolution of Industrial Districts, Kluwer, Boston (2003)
44. Spohrer, J., Golinelli, G.M., Picocchi, P., Bassano, C.: An integrated SS-VSA analysis of changing jobs. Service Science 2(1/2), 1–20 (2010), ISSN 978-1-4276-2090-3
45. Coase, R.: The Nature of the Firm. Economica 4, 386–405 (1937)
46. Cohen, W., Levinthal, D.: Innovation and learning: the two faces of R&D. Economic Journal 99 (1989)
47. Enright, M.: Regional Clusters and Firm Strategy. In: Chandler, A., Hagstrom, P., Solvell, O. (eds.) The Dynamic Firm: The Role of Technology, Strategy, and Regions. Oxford University Press, Oxford (1998)
48. Hannah, M., Freeman, J.: Organizational Ecology. Harvard University Press, Cambridge (1989)
49. Harvard Business School: Cluster Mapping Project, Institute for Strategy and Competitiveness. Harvard Business School, Cambridge (2002)
50. Piore, M., Sabel, C.: The Second Industrial Divide: Possibilities for Prosperity. Basic Books, New York (1984)
51. Porter, M.: Clusters and the New Economics of Competition. Harvard Business Review 76, 77–90 (1998)

Views on Scientific Workflows

Mirko Sonntag, Katharina Görlach, Dimka Karastoyanova, Frank Leymann, Polina Malets, and David Schumm

Institute of Architecture of Application Systems, University of Stuttgart, Universitaetsstrasse 38, 70569 Stuttgart, Germany
`{sonntag,goerlach,karastoyanova,leymann,malets,schumm}@iaas.uni-stuttgart.de`

Abstract. Workflows are becoming more and more important in e-Science due to the support they provide to scientists in computer simulations, experiments and calculations. Our experiences with workflows in this field and the literature show that scientific workflows consist of a large number of related information. This information is difficult to deal with in a single perspective and has changing importance to scientists in the different workflow lifecycle phases. In this paper we apply viewing techniques known from business process management to (service-based) scientific workflows to address these issues. We describe seven of the most relevant views and point out realization challenges. We argue that the selected views facilitate the handling of workflows to scientists and add further value to scientific workflow systems. An implementation of a subset of the views based on Web services and BPEL shows the feasibility of the approach. The presented work has the goal to increase additionally the acceptance of the workflow technology in e-Science.

Keywords: Process views, BPEL, Web services, SOA, simulation workflows, scientific workflows.

1 Introduction

In the last years the application of workflows in scientific simulations and scientific computing has been experiencing an increased attention [1]. In this context workflows have been used to successfully implement, for instance, image processing tasks in physical astronomy [2], earthquake simulations in geology [3], or calculations related to biodiversity of species [4]. There are many reasons why workflows are interesting for natural or engineering scientists: (1) simulations often consist of manual steps that can be automated with the help of workflows; (2) former monolithic (legacy) scientific applications can be executed on multiple machines in a distributed manner; or (3) new simulations/calculations can be created in a graphical manner by modeling workflows, i.e. the programming effort is decreased. The main goal is to allow and facilitate scientists to concentrate on their core competencies and research topics instead of coping with IT issues.

Besides these technical improvements that workflows provide to scientists a user-friendly handling is a key concern in scientific workflow management (WFM). Workflows in e-Science possess many aspects that are interesting for scientists but

J. Grabis and M. Kirikova (Eds.): BIR 2011, LNBIP 90, pp. 321–335, 2011.

hard or even impossible for a human to capture in a single perspective. Scientific workflows can consist of hundreds of activities (e.g. as described in [5]) with varying degree of importance for scientists. Scientists need to have a look at the data flow between activities to see the logical dependencies between tasks. In some cases it is required to see at a glance which tools where executed on which machines in what time. These and other information can be prepared and clarified with the help of viewing techniques for scientific workflow, which filter out unnecessary information.

While process views are a well-known technology in business WFM [6][7][8], a detailed investigation of their applicability to scientific workflows has not been carried out. Most scientific workflow management systems (SWFMS) are built from scratch and do not rely on the conventional workflow technology typical for business applications [9][10]. In the scope of the Stuttgart Research Center Simulation Technology (SRC SimTech[1]) we develop a SWFMS based on the traditional workflow technology in a SOA environment. In our work with different scientific institutes we have conducted case studies in which we have gathered requirements from scientific researchers and implemented scientific simulations with workflows and Web services (WSs) (e.g. [11]). In this paper we focus on process views that are useful to visualize the different perspectives of scientific workflows to scientists, i.e. the simulation or experiment itself and not the results of the experiment/simulation. For the description of the perspectives we build on previous work in the field of process view application scenarios [12] and viewing techniques [13]. In our former work we have developed a lifecycle definition of scientific workflows that reflects the iterative and adaptive development of scientific workflows [14] (see also Figure 1). Due to this different lifecycle it is needed that the existing viewing mechanisms and techniques are adapted to the needs of scientists and scientific applications. We selected seven views most relevant to the everyday work of scientists. These views mainly concern instance monitoring, process analysis, and abstraction. Views for process re-use have been identified important for scientific workflows but have already been proposed in former works on process view transformations [15][16][17]. To describe the views we make use of a workflow for the simulation of the ink diffusion in a glass of water as running example. We are convinced that the process views we present here add value to SWFMSs.

The paper contributes (1) an advancement of the current state-of-the-art of scientific WFM by exploiting viewing techniques that have been identified recently in the field of traditional process management; (2) extensions and refinements to existing process views that meet different requirements of scientific computing; and (3) a proof of the concept by the implementation of a subset of the proposed views in a SWFMS based on BPEL [18] and WSs.

The rest of the paper is structured as follows. Section 2 discusses related work in the field of process views and views in existing SWFMSs. Section 3 presents a collection of views on scientific workflows and provides details about the extensions that were necessary. Section 4 shows a prototypical implementation of a subset of the views of Section 3. Section 5 closes the paper with conclusions and an outlook.

---

[1] `http://www.simtech.uni-stuttgart.de`

2 Related Work

In business process management (BPM), viewing techniques are currently gaining momentum. The increasing complexity of business processes requires the use of advanced abstraction and visualization techniques. Viewing approaches for the omission and aggregation of tasks [19] as well as those related to analysis, monitoring and graphical display [12][13] are relevant to our work. Process monitoring using viewing techniques has been thoroughly investigated (e.g. in [20]). These monitoring views can also be applied to ease work in complex scientific workflows as we discuss in Section 3.3. These different views and the concepts behind them need extensions in order to fit the needs of scientists and scientific applications.

In [21] Petre argues that for different groups of people, different graphical notations, icons, shapes and so on needs to be provided to account for different understanding. While some icons used in business process automation frameworks might be universally applicable, in scientific workflows other shapes might be useful to ease understanding. We take a first step into this direction by proposing the use of custom icons for scientific computation services as discussed in Section 3.5.

Cohen-Boulaki et al. [22] used viewing techniques for different levels of granularity and abstraction to solve the provenance challenge. They demonstrated that such techniques are well applicable to support reasoning about all the intermediate and final data produced in the course of execution of scientific workflows.

Existing SWFMSs also make use of views. In e-BioFlow[2] it is possible to switch between control and data flow perspectives. The current status of the workflows is displayed in a table. Kepler[3] and Triana[4] allow the modeling of complex activities that hide more complex workflow logic from the users. Both make use of a monitoring view to display the runtime status of workflows. Taverna[5] also provides a view for the status of running workflows. Pegasus[6] contains a monitoring component for the analysis of past workflow runs.

3 Views on Scientific Workflows

Scientists and their applications impose new requirements on workflow systems, e.g. data-centricity, tool integration, hiding of technical details, clear arrangement of workflows when applied in the field of scientific research [9][10]. Most of these requirements can be met by existing process views from conventional workflow technology [13]. However, in our former work we have observed that the lifecycle phases modeling, execution, monitoring and adaptation of conventional workflows are alternating and continuously repeating in scientific workflows, i.e. scientific workflows are developed in a trial-and-error approach (see Figure 1) [14]. The reason is that the direction of an experiment may not be predictable and hence an adaptation

---

[2] `http://sourceforge.net/projects/e-bio-flow/`
[3] `https://kepler-project.org/`
[4] `http://www.trianacode.org/`
[5] `http://www.taverna.org.uk/`
[6] `http://pegasus.isi.edu/`

of running experiments is required. This and the fact that scientists play all roles participating in the workflow lifecycle motivate the need for an integrated tool that supports scientists in all lifecycle phases. A modeling tool for scientific workflows hence has to be able to also steer workflow execution, to monitor workflows and to adapt running workflows. Some of the process views therefore need extensions to be applicable for scientific workflows. The views are not automatically derived visualizations of aspects of a scientific workflow in another tool. They are part of the modeling tool and thus can be used to model scientific workflows. Another reason for extensions is that information about the computing infrastructure is of interest to scientists, e.g. properties of the employed servers.

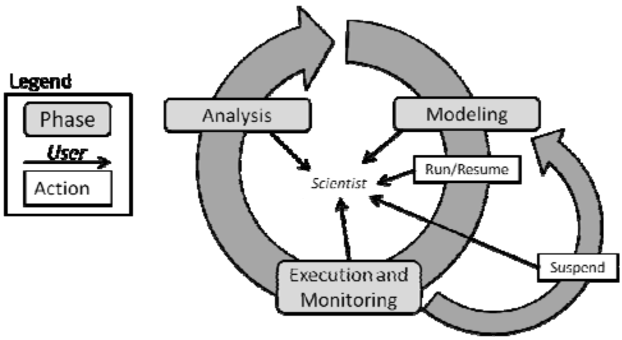


Fig. 1. Definition of the lifecycle of scientific workflows [14]

We have chosen a scientific workflow for the simulation of ink diffusion in a glass of water over a period of time as running example. Core of the simulation is Dune[7], a C++ toolbox to solve partial differential equations (PDEs) with the help of grid-based methods (e.g. finite elements method (FEM), finite volumes). Note that we (and the Dune community) understand the term "grid" (also called mesh) here as a graph of nodes and edges used for complex calculations and not as a computer infrastructure. We implemented this simulation with BPEL to orchestrate DUNE services; the DUNE services are WSs providing Dune modules for remote use in a network. For the purpose of legibility we created a BPMN [23] representation of this BPEL workflow with the most important steps only (see Figure 2).

The ink diffusion in water simulation consists of four main phases. Firstly, a new simulation instance in the Dune framework is created (steps 1-6). The input parameter file is unpacked; the parameters are inserted into the Dune/simulation source code which then gets compiled. Secondly, the raw grid is created that describes the glass of water as graph of nodes and edges (step 7). This grid is the discretization of the glass of water. Then, the grid is refined by multiplying the nodes and edges with the goal to gain much more detailed results (steps 8-9). Thirdly, the simulation is conducted on the refined grid (step 10). Multiple iterations are needed to simulate how the ink distributes in the water. Each loop step represents a simulation time step. Finally, the simulation is stopped and the simulation instance is closed (steps 11-13).

[7] Distributed and Unified Numerics Environment, `http://www.dune-project.org`

In the rest of the section we present selected views. All views are shown in the same way by giving a motivation, listing prerequisites, describing the approach itself including a figure and considering challenges.

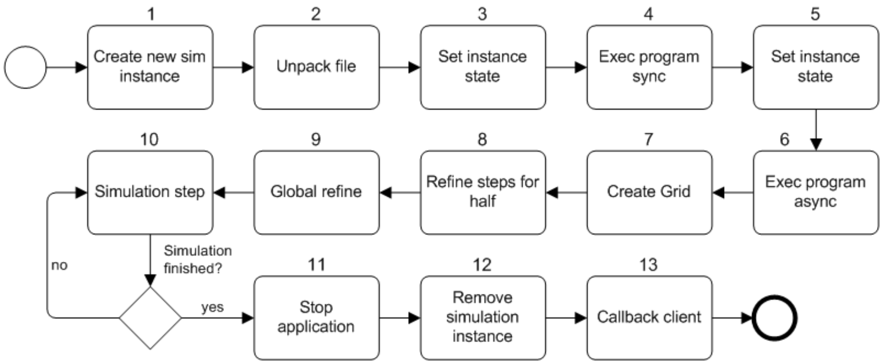


Fig. 2. BPMN diagram of the scientific workflow for simulation of ink diffusion in water. All activities denote service invocations. All other activities are omitted for the reason of legibility.

3.1 Aggregation of Complex Workflow Logic

Motivation. Scientific workflows often implement complex logic that can be divided in different functional parts. For example steps 7-9 in Figure 2 create and refine an FEM grid needed for the subsequent simulation steps. Such a self-contained logic/workflow fragment implements well-known behavior. It can comprise a huge number of activities and may have multiple entries and exits.

Figure 3a shows another example, a fragment for the robust allocation of a scientific service from a resource manager for a computing intensive task (e.g. a PDE solver service). If service allocation fails, the service request is retried a predefined number of times. The fragment implements useful behavior, but the workflow's legibility suffers from its complexity (especially for non-computer scientists). A view is needed to aggregate selected complex workflow behavior so that it looks like a single entity and the scientists can focus on the relevant experiment's logic.

Prerequisites. The subset of aggregated activities should be restricted to a connected set of activities. This prevents from ambiguity and cycles in the workflow graph when aggregating the activities.

Approach. Aggregation views in BPM are usually automatically derived by certain algorithms. In scientific WFM the scientist manually selects a number of activities in a workflow model that ought to be aggregated. This provides maximal flexibility for a customization that fits the needs of a scientist. A transformation step then automatically translates the workflow model into a (graphical) representation where the selected activities are replaced by a single activity that now stands for the complete behavior of the aggregate. Note that BPMN has a built-in mechanism to represent aggregates: collapsed activities. The aggregation view on the fragment "robust allocation of a service" is shown in Figure 3b. In contrast to outsourcing workflow logic with the help of sub-workflows the aggregation of logic effects only the modeling and not to the runtime of workflows. In particular it is possible to gain insight into the aggregated workflow logic in a straight-forward way because the logic is still part of the workflow model. The aggregation view represents logic as a black box where the implementation details can be accessed on demand. As opposed to this, the logic of sub-workflows is not visible in the parent/invoking workflow.

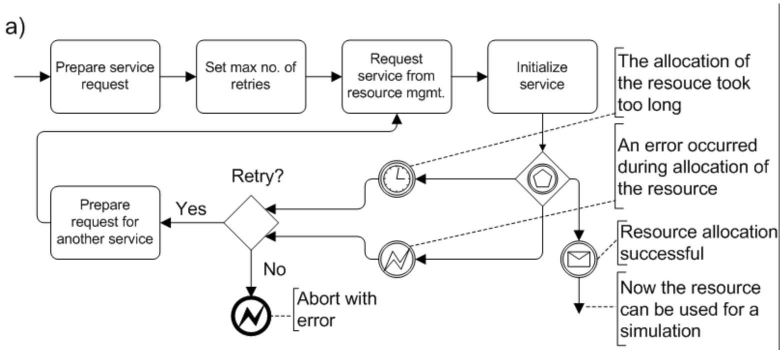


Fig. 3. Process fragment for robust service allocation (a) and the aggregation view on the same fragment (b)

Challenges. The main difficulty is the replacement of selected activities if the aggregate has multiple entries or exits. Another challenge is an intelligent visualization to enable expanding and collapsing of the aggregates.

3.2 Phases in Simulation Workflows

Motivation. In our work with scientists we have implemented several simulation use cases with WSs and workflows (e.g. [11]). Usually the simulations follow a simple pattern of phases: (1) the pre-processing comprises data and simulation preparation such as parameter specification, data import, FEM grid generation, or directory setup; (2) the actual simulation is often a loop of resource-demanding calculations, e.g. solving a PDE; (3) during the post-processing phase result data is visualized and the simulation environment is cleaned up (e.g. de-allocation of resources, deletion of intermediate files). Although a simulation consists in principle of these phases, a simulation workflow can be huge and complex. The phases consist of more than a simple sequence of service calls. Fault handling, robust service allocation (see Figure 2a) and usage, as well as transaction logic can convolute the workflow logic so that scientists can lose the overview. A view on the workflow that reflects the simulation phases is deemed useful.

Prerequisites. The phases in a simulation workflow should be detectable and in sequential order. Otherwise a simulation could conduct a step back, e.g. from simulation to pre-processing, contradicting the common understanding of the phases.

Approach. The view on the simulation phases can be created automatically or manually. Automatic generation is based on a mapping of activities on phases. A mapping could, e.g., be geared to the usage of specific services (e.g. an activity for the invocation of a visualization service belongs to the post-processing phase) or activity names (e.g. an activity "solve PDE" belongs to the simulation phase). For the manual creation of the view the scientist selects a number of activities and assigns them to a phase. Not all simulations have to be mapped on the three afore-mentioned phases. It may be required to introduce finer grained phases, e.g. post-processing for intermediate results. Therefore it should be possible for scientists to customize the simulation phases that the workflow modeling tool provides (e.g. reordering, renaming). After the mapping of the workflow logic on phases is complete (Figure 3a), a transformation step translates the workflow model with the help of an aggregation view into a workflow that only consists of the resulting phases as aggregates (Figure 3b). Then, the phases are illustrated as arrows that prescribe their sequential ordering (Figure 3c). The implementation details of the simulation are hidden to the scientist but can be requested on demand (similar to the aggregation of complex workflow logic in Section 3.1).

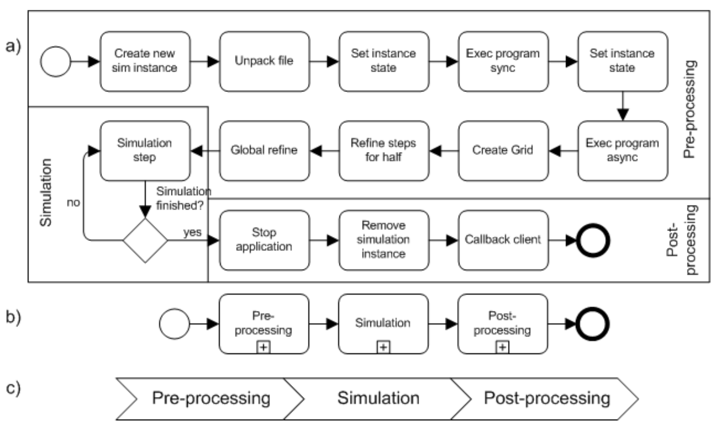


Fig. 4. Aggregation view on a simulation workflow to show only the high-level simulation phases. The simulation workflow is mapped on simulation phases (a). An aggregation view on the activities of the phases reduces the complexity of the model (b). The aggregates are then visualized as arrows (c).

Challenges. The main challenge is to find an appropriate visualization of the simulation phases if they are interleaved to a certain degree.

3.3 Status of a Scientific Workflow

Motivation. For the trial-and-error scientific workflow development it is needed to enrich workflow models in the modeling tool with information about the execution status of workflow instances. Scientists can then monitor the progress of their simulations while still modeling them—a requirement completely new to business workflows where modeling and monitoring are accomplished by different tools.

Prerequisites. In order to visualize the runtime status of workflow instances in a modeling tool the tool needs access to instance data. This can be achieved e.g. by querying the audit trail or by listening to execution events that are published by the employed workflow engine. Note that the used technique strongly depends on the API the workflow engine offers. Another prerequisite is that the workflow model of the instance a scientist wants to monitor is available in the modeling tool. Otherwise the instance state cannot be monitored.

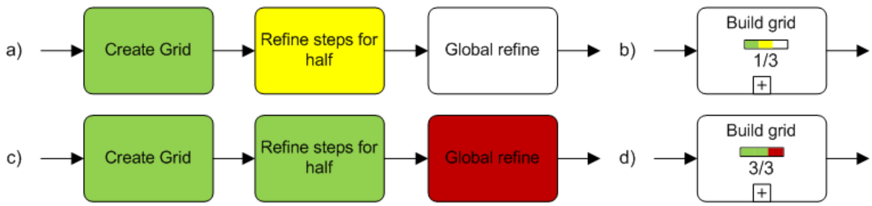


Fig. 5. Workflow model with annotated runtime information. Completed activities are green (a, c), running activities yellow (a), and failed ones red (c). Aggregated activities must have a visualization that reflects the state of their contained activities, e.g. by a multi-colored status bar (b, d).

Approach. The current states of the instantiated workflow model elements are attached to the workflow model as well as other information that is relevant for the visualization of a workflow instance (e.g. number of completed iteration in loops). The illustration of the workflow model is changed according to the augmented runtime information. Instance states are mapped to colors; these colors are used to display activities (see Figure 5a and 5b). In loops it is useful to visualize the number of passed loop iterations. Special attention has to be paid to aggregations. The status of an aggregation depends on the states of all contained activities. It is thus required to calculate the aggregation state and/or to filter information not needed to illustrate the instance state (e.g. state changes of activities in an aggregation). The state of an aggregation could be displayed with the help of a status bar (see Figure 5b and 5d).

Challenges. The main difficulty is how to bridge the separation between workflow models and instances. The question is how to visualize different instances in a modeling tool that is inherently unaware of workflow instances. The tool needs an extension to allow users to select the workflow instance to monitor. Selection of instances should be based on given or generated meta-data or the starting time instead of instance IDs. Nevertheless, the instance IDs of the currently monitored instances are important to correlate runtime information to the monitored instances.

3.4 Data Flow Visualization

Motivation. Current languages in conventional workflow technology are control flow-oriented, e.g. BPEL, BPMN. In contrast, existing SWFMSs usually follow a data flow-oriented modeling paradigm [1][24]. The reason is that scientists think in a data-oriented way. This fact creates the need for an explicit visualization of data aspects in control flow-oriented workflow languages when being used for scientific applications.

Prerequisites. The size of data items can be calculated not until runtime.

Approach. The workflow languages used in the conventional workflow technology usually provide means to implicitly specify the flow of data by means of variables. Based on this implicit data flow a transformation step can calculate the explicit data dependencies between two activities or between an activity and a data source or sink (e.g. variables, databases, files) (Figure 6). Two activities A and B have a data dependency if activity A produces data that is used as input for activity B or vice versa. After the transformation step the data links have to be visualized. We recommend views that exclusively illustrate control dependencies, exclusively data dependencies, or both combined. That way the scientist can select the view that best displays the detail he is interested in (e.g. the control flow view is best for displaying loops, the data flow view can help optimize a workflow by task parallelization). Another aspect of a data flow view is to visualize the amount of data that is transferred within the workflow and between used services. This information is useful to reduce expensive data transfer operations. The amount of data can be represented by the data link color or width (Figure 6). The data size has to be collected at runtime. The generated data links of the workflow model have to be augmented with this information and the illustration of the data links can be adapted accordingly.

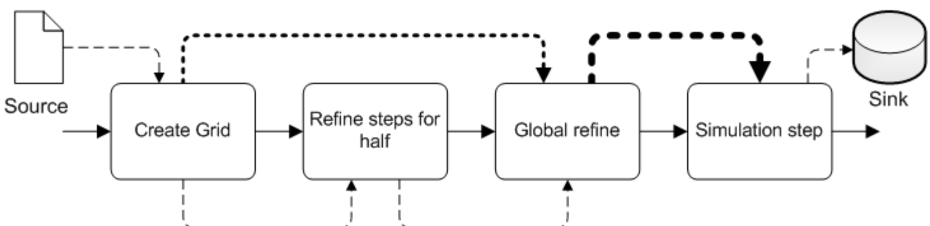


Fig. 6. View for explicit data flow between activities and sources/sinks by dotted arrows. The arrow width is an indicator for the data size.

Challenges. Determining the size of data is not straight-forward because many data objects exist only in the memory of the workflow system. If data is passed by reference it has to be specified if the reference itself or the referenced data counts for the data size. Techniques have to be developed to determine the size of data that is external to the workflow engine (e.g. files).

3.5 Custom Icon for Service Invocation

Motivation. One of the most important requirements on sWfMSs is usability since the majority of users of sWfMSs are no computer scientists. A scientist needs the support of easy-to-use tools and self-explaining visualizations in these tools. Scientific workflows can encompass a lot of service invocations and other activities. Existing workflow tools (e.g. the Eclipse BPEL Designer) or graphical workflow languages (e.g. BPMN) foresee the visualization of activities on per activity-type-basis, i.e. all activities of a certain type have the same icon. Activities are usually customized by their names. This makes it difficult and time-consuming to orient oneself in a forest of nodes, edges, labels and recurring icons. The idea is to allow customization of activity icons in the workflow modeling tool to facilitate orientation in a workflow graph by different visual symbols.

Prerequisites. The customization has to be done with mnemonic and self-explanatory icons to be of real value to scientists.

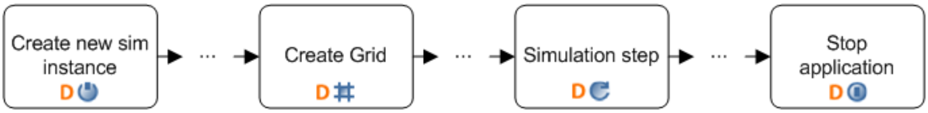


Fig. 7. Customized icons for service invocations of the Dune-library

Approach. Customization of activity icons can be done manually or automatically. In the manual case the scientist simply selects an activity and specifies a new icon. The modeling tool should provide icons for standard cases (e.g. different sensors, databases, visualization applications such as Gnuplot). Additionally, it should be possible to import and select custom icons. In the automatic case a mapping of activities on icons is needed. This can be realized by an external file that maps URLs of icons on activities by certain criteria (e.g. the invoked service operation, the location of the invoked operation, the service binding, the activity's name). Another option is a WSDL extension where the URL of an icon is attached to a WSDL operation. In a workflow modeling tool the icon can then be loaded and used for the activity that calls this operation. Figure 7 shows the service calls of Dune-library in our "ink diffusion" example with customized icons.

Challenges. The WSDL extension for operation icons entails additional work to service providers. They have to create an icon and publish it together with the service. The workflow modeling tool has to be able to deal with the icon's format. The icon's size should have an upper boundary because it has to be downloaded during workflow modeling. Problems arise when the computer that is used to model the workflows does not have an internet connection or the service is temporarily down. In this case the standard icon could be taken until the predestined icon is available.

3.6 Performance Analysis

Motivation. In the analysis and optimization of scientific workflows, time plays a crucial role (e.g. the time particular resources are occupied, the time needed to transfer large amounts of data, the runtime of (parts of) the workflow execution). The resource consumption is another determinant of performance. The idea is to build a view that makes the performance perspective of a workflow instance visible to a scientist in the workflow design environment. The impact of concrete machines to the workflow throughput is not considered in business process views where the servers are transparent to the workflow.

Prerequisites. Firstly, the "effective" (executed) workflow model has to be constructed that might differ from the originally designed workflow model due to adaptation steps (e.g. automatic insertion of data transfer activities). For this construction we can benefit from process mining tools and techniques (e.g. the ProM framework [25]). Secondly, the effective workflow model has to be imported into the modeling tool and augmented with runtime information about the time and resource consumption aspects of the workflow execution (e.g. the duration of activities/service invocations, the duration of data transfers, consumed processing units (e.g., measured in FLOPS)). This information needs to the annotated to the activities contained in the effective model.

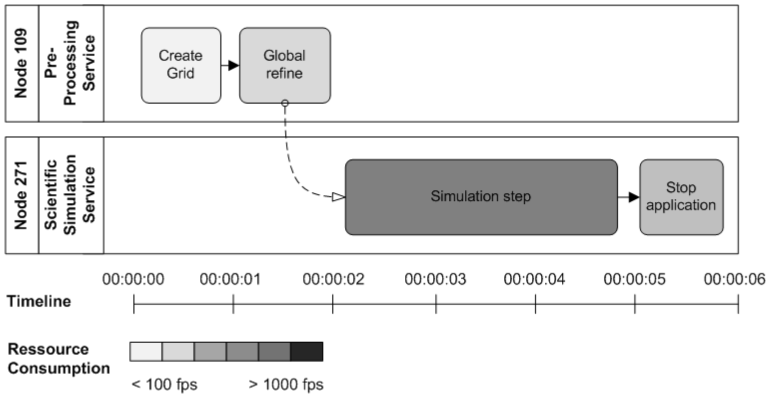


Fig. 8. Performance analysis is supported through stretching long-running activities and arranging them along a timeline. The degree of resource consumption is visualized by using different levels of grey for activity shapes.

Approach. As a first step, we unfold loops to clearly arrange the number of iterations, to identify problematic branches and differences in the execution of iterations. As a result, we obtain a complex workflow model containing possibly hundreds or thousands of activities. As a consequence, we need to apply abstraction techniques to simplify this model. This encompasses removing dead and unreachable paths which do not have an impact on the performance. Further, we can omit all activities which are in terms of duration or resource consumption below a certain threshold that needs to be specified on a per case basis. The resulting model is then layouted and displayed. We propose to arrange the activities along a timeline, and stretch their shapes in order to express the time dimension. To distinguish between the different computing nodes and the used scientific services it is meaningful to apply pools and swim lanes. The consumption of computing resources can be visualized by coloring the activities (see Figure 8).

Challenges. For the measurement of resource consumption in used servers these servers have to be instrumented accordingly. Another challenge is the mapping of the performance data to activities of the effective model. Finally, creating a well-readable layout is not trivial for long-running, complex scientific workflows that orchestrate services on lots of nodes.

3.7 Access to Runtime Information of Used Services

Motivation. Besides the workflow instance status information during runtime (see Section 3.3) the status of used services and resources as well as an insight into used and produced data is also of interest. This makes the simulation infrastructure visible and gives scientists the full control over their experiments and simulations. In contrast to this, in BPM scenarios the used machines are invisible.

Prerequisites. The used services need to provide operations that allow querying information about the machines they are installed on as well as information about used and produced data.

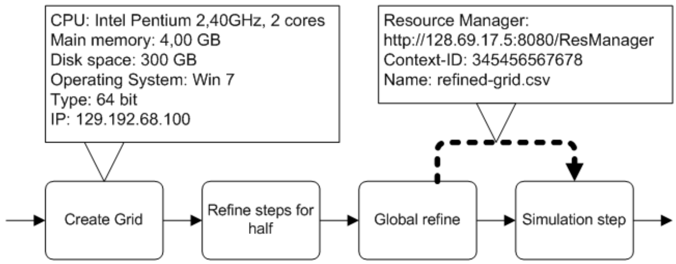


Fig. 9. View on service and data information

Approach. This view collects and visualizes simulation instance data beyond the state of workflow activities. This includes information about the machines scientific services are running on (e.g. CPU, storage), the content of variables (or data passed via data links) in the workflow and data external to the workflow (e.g. files, databases) (see Figure 9). It is important to mention that this view does not augment the workflow model with the collected information. In fact the data should be accessed on demand by the scientist because it can encompass huge amounts of data (e.g. result files of hundreds of mega bytes). The data is distributed in the scientific workflow infrastructure and only loaded to the workflow modeling tool on explicit request. After that it is visualized in a user-friendly way, e.g. by a pop-up.

Challenges. Many different data sources (e.g. resource manager, audit trails, servers) and representations (e.g. files, databases) have to be integrated, processed and visualized. Since the data is spread over the infrastructure it may be difficult to find data related to a particular simulation run. Sophisticated correlation mechanisms have to be used (e.g. a global context ID for each simulation run that is attached to all messages and hence known to all participating services).

4 Implementation

We are developing a SWFMS based on the WS technology and BPEL. The idea is to introduce the numerous advantages of conventional workflows to e-Science [24], e.g. robust workflow execution, fault handling and transactions on the workflow level, asynchronous messaging. But the technology needs thorough extensions to become interesting for scientists, e.g. support for trial-and-error workflow development (see lifecycle in Figure 1), integration of stream data and sensors, or data as first class citizen. Our current prototype of the SWFMS implements some of these extensions and a subset of the views presented in this paper. The latter is subject of this section.

The prototype is based on the Apache Orchestration Director Engine (ODE) as BPEL engine, the Eclipse BPEL Designer as modeling tool and Apache Active MQ as message queuing system. A demonstration of the prototype is available in [26].

We extended the visualization of `invoke` activities to implement the view *custom icon for service invocation*. If an invoke activity gets configured with a partner link and an operation of a WSDL, then the modeling tool looks up whether an icon is registered with this WSDL operation. If so, the icon is fetched and used for this invoke activity (Figure 10a, activity "Global Refine"). Otherwise the standard icon of the modeling tool is taken (Figure 10a, e.g. activity "Create Grid"). The custom icons are attached as URI to WSDL operations by the new attribute `icon` (see Listing 1).

```
<wsdl:definitions
    xmlns:icon="http://iaas.uni-stuttgart.de/wsdlIconExtension">
  <operation name="thisOp"
      icon:icon="http://exampleService/thisOp/icon.gif">...
  </operation>
</wsdl:definitions>
```

Listing 1. WSDL extension for custom operation icons

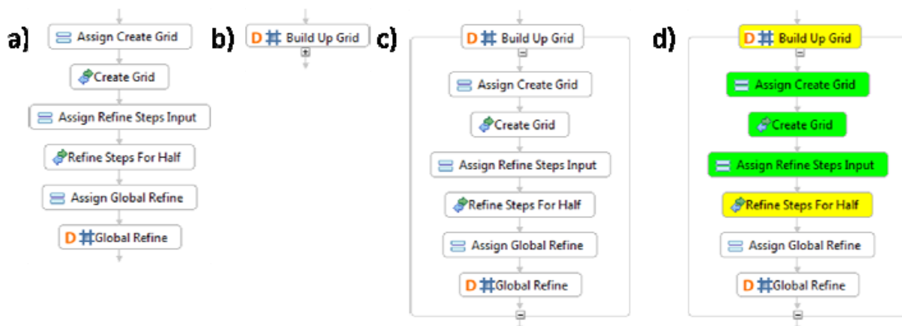


Fig. 10. Eclipse BPEL Designer extension for custom icon for service invocation (a), aggregation collapsed (b) and expanded (c), and monitoring (d)

The *aggregation* view is in parts already supported by the Eclipse BPEL Designer: structured activities can be collapsed so that they appear as a single basic activity. We extended this mechanism so that a set of consecutive activities can be marked by a user and aggregated explicitly via the context menu. The aggregated activities are put into a collapsed sequence (Figure 10b). The sequence activity inherits the icon of the last invoke activity of the aggregate. The user can expand the activity to gain insight into the aggregate's logic (Figure 10c). Vice versa it is also possible to disaggregate the activities via a context menu command.

Finally, the *status of a scientific workflow* view is realized as follows. The BPEL Designer is extended so that scientists can start the opened workflow from within the tool. The workflow model is then deployed on an engine and a new instance is started. The extended engine publishes execution events over a topic. The modeling tool subscribes to this topic, receives these events, correlates them to the opened model and colors activities according to the received activity instance state (Figure 10d).

5 Conclusions

Scientific workflows possess properties that make it difficult to handle them during modeling, monitoring and analysis (e.g. size of the workflows, data and data flow as first class citizen, used resources). In this paper, we have shown how process viewing techniques can be used to ease dealing with workflows to scientists and to reveal potentials for the optimization of workflow execution. Process views are well elaborated on in BPM. In the context of scientific workflows, however, an investigation of their applicability was missing. We have filled this gap by presenting the concept of seven views relevant to scientists and scientific workflows. We are convinced that the implementation of these views adds value to any SWFMSs. As a proof of concept we have implemented a subset of the proposed views. Some of described concepts could also be generalized and provided back to BPM (e.g. custom icons, performance analysis).

In BPM, efforts were made to find icons for recurring actions in business processes [27]. With the implementation of the view *custom icon for service invocations* and the associated WSDL extension we have developed the technical basis for a similar approach in scientific WFM. Further research will show whether frequently recurring actions can be identified and standard icons can be found. In future we want to implement the missing views and find more views, e.g. for stream data or security.

Acknowledgements. The authors would like to thank the German Research Foundation (DFG) for financial support of the project within the Cluster of Excellence in Simulation Technology (EXC 310/1) at the University of Stuttgart.

References

1. Taylor, I., Deelman, E., Gannon, D., Shields, M.: Workflows for e-Science: Scientific Workflows for Grids. Springer, Heidelberg (2007)
2. Berriman, G.B., Deelman, E., Good, J., et al.: Generating Complex Astronomy Workflows. In: Taylor, I., et al. (eds.) Workflows for e-Science. Springer, Heidelberg (2007)
3. Maechling, P., Deelman, E., Zhao, L., et al.: SCEC CyberShake Workflows—Automating Probabilistic Seismic Hazard Analysis Calculations. In: Taylor, I., et al. (eds.) Workflows for e-Science. Springer, Heidelberg (2007)
4. Jones, A.: Workflow and Biodiversity e-Science. In: Taylor, I., et al. (eds.) Workflows for e-Science. Springer, Heidelberg (2007)
5. Deelman, E., Mehta, G., Singh, G., et al.: Pegasus: Mapping Large-Scale Workflows to Distributed Resources. In: Taylor, I., et al. (eds.) Workflows for e-Science. Springer, Heidelberg (2007)
6. Eshuis, R., Grefen, P.: Constructing Customized Process Views. Data & Knowledge Engineering 64(2), 419–438 (2008)
7. Zhao, X., Liu, C., Sadiq, W., et al.: Implementing Process Views in the Web Service Environment. World Wide Web Journal 14(1), 27–52 (2011)
8. Polyvyanyy, A., Smirnov, S., Weske, M.: Business Process Model Abstraction. In: Dayal, U., Eder, J., Koehler, J., Reijers, H.A. (eds.) BPM 2009. LNCS, vol. 5701. Springer, Heidelberg (2009)
9. Sonntag, M., Karastoyanova, D., Leymann, F.: The Missing Features Of Workflow Systems For Scientific Computations. In: Proc. of the 3rd Grid Workflow Workshop (GWW), GI-Edition LNI, Volume P-160 (2010)

10. Sonntag, M., Karastoyanova, D., Deelman, E.: Bridging The Gap Between Business And Scientific Workflows. In: Proc. of the IEEE 6th Int. Conf. on e-Science. IEEE Press, Los Alamitos (2010)
11. Sonntag, M., Hotta, S., Karastoyanova, D., Molnar, D., Schmauder, S.: Workflow-Based Distributed Environment For Legacy Simulation Applications. In: Proc. of the 6th Int. Conf. on Software and Data Technologies (ICSOFT 2011), Poster Presentation (2011)
12. Schumm, D., Anstett, T., Leymann, F., Schleicher, D.: Applicability of Process Viewing Patterns in Business Process Management. In: Proc. of the Int. Workshop on Models and Model-driven Methods for Service Engineering (3M4SE 2010). IEEE Press, Los Alamitos (2010)
13. Schumm, D., Leymann, F., Streule, A.: Process Viewing Patterns. In: Proc. of the 14th IEEE International EDOC Conference (EDOC 2010). IEEE Computer Society Press, Los Alamitos (2010)
14. Sonntag, M., Karastoyanova, D.: Next Generation Interactive Scientific Experimenting Based On The Workflow Technology. In: Proc. of the 21st IASTED Int. Conf. on Modelling and Simulation. Acta Press (2010)
15. Goderis, A., Sattler, U., Lord, P., Goble, C.A.: Seven Bottlenecks to Workflow Reuse and Repurposing. In: Gil, Y., Motta, E., Benjamins, V.R., Musen, M.A. (eds.) ISWC 2005. LNCS, vol. 3729, pp. 323–337. Springer, Heidelberg (2005)
16. Avrilionis, D., Cunin, P.Y., Fernström, C.: OPSIS: A View Mechanism for Software Processes Which Supports Their Evolution And Reuse. In: Proc. of the 18th Int. Conf. on Software Engineering (ICSE). IEEE Computer Society, Los Alamitos (1996)
17. Schumm, D., Leymann, F., Streule, A.: Process Views to Support Compliance Management in Business Processes. In: Proc. of the 11th Int. Conf. on Electronic Commerce and Web Technologies (EC Web). Springer, Heidelberg (2010)
18. Organization for the Advancement of Structured Information Standards (OASIS): Business Process Execution Language 2.0, BPEL (2007)
19. Bobrik, R., Reichert, M., Bauer, T.: View-Based Process Visualization. In: Alonso, G., Dadam, P., Rosemann, M. (eds.) BPM 2007. LNCS, vol. 4714, pp. 88–95. Springer, Heidelberg (2007)
20. Reichert, M., Bassil, S., Bobrik, R., Bauer, T.: The Proviado Access Control Model for Business Process Monitoring Components. Enterprise Modelling and Information Systems—An International Journal 5(3), 1860–6059 (2010) ISSN 1860-6059
21. Petre, M.: Why Looking Isn't Always Seeing: Readership Skills and Graphical Programming. Communications of the ACM 38(6) (1995)
22. Cohen-Boulakia, S., Biton, O., Cohen, S., Davidson, S.B.: Addressing the Provenance Challenge using ZOOM. In: Concurrency and Computation: Practice and Experience, vol. 20 (2008)
23. Object Management Group (OMG): Business Process Model and Notation (BPMN). OMG Available Specification, Version 2.0, Beta 1 (2009)
24. Goerlach, K., Sonntag, M., Karastoyanova, D., et al.: Conventional Workflow Technology for Scientific Simulation. In: Yang, X., Wang, L., Jie, W. (eds.) Guide to e-Science. Springer, Heidelberg (2011)
25. Process Mining Group. ProM (2011), `http://prom.win.tue.nl/tools/prom/`
26. Sonntag, M.: Views on scientific workflows. Prototype demo (2011), `http://www.iaas.uni-stuttgart.de/institut/mitarbeiter/sonntag/indexE.php#viewsVideo`
27. Mendling, J., Recker, J., Reijers, H.A.: On The Usage Of Labels And Icons In Business Process Modeling. Int. Journal of Information System Modeling and Design (IJISMD) 1(2) (2009)

Agile Business Process Management in Research Projects of Life Sciences

Silke Holzmüller-Laue[1] and Bernd Göde[2]

[1] Center for Life Science Automation (celisca), University of Rostock, F.-Barnewitz-Str. 8, 18119 Rostock, Germany
[2] Institute of Automation, University of Rostock, R.-Wagner-Str. 31, 18119 Rostock, Germany
{silke.holzmueller-laue,bernd.goede}@uni-rostock.de

Abstract. In life science laboratories the sub-process automation of methods with semi- or full automated, isolated solutions called islands of automation and several IT systems dominate. There are deficits in networking of these sub-processes. The R&D processes in the life sciences research are complex, flexible, unstructured, knowledge-intensive, distributed, and parallel; they use heterogeneous resources, and combine automated, semi-automated, and manual activities in high-variable process chains with a high number of control structures. This characteristic of LSA-processes makes high demands on an integrated process management that contains an interdisciplinary collaborative process control and the documentation of the global process from for example purchasing, sample storage, method development to analytics and interpretation of results as well as the extraction of knowledge.

Using methods, techniques, and tools of business process management (BPM) in the life science automation offers the potential to improve the automation level, the networking and the quality of the life science applications. The authors investigate the suitability of the standard based methods and techniques of BPM for the introduction of a flexible, integrated, and automated process management in the heterogeneous and hybrid systems in life sciences. This article will be focusing on the advances made in distributed workflow automation in highly variable system and application environments of research projects by use of BPM methods and tools.

Keywords: Agile R&D Processes, BPM, Life Sciences, Process Automation, Workflow Management.

1 Introduction

The aim of use of BPM method and tools in the life science automation is an effective automation of the overall processes in fast changing heterogeneous R&D environments and to reduce the developer effort significant. The topic of this paper is to present the idea, the concept, the challenges and the advantages of using business process management methods and tools in the life science automation as well as first results and experiences.

J. Grabis and M. Kirikova (Eds.): BIR 2011, LNBIP 90, pp. 336–344, 2011.

1.1 Challenges in Life Science Automation

Flexible laboratory automation is essential in the most research projects in life sciences. In addition to time constraints and economic reasons, a further goal of automation in pharmaceutical and biotech companies or institutes is the improvement of experimental results and workflow [1]. Research and development in the life science automation have been working with semi- and fully automated islands of automation systems (laboratory robot systems, chemical parallel reactors, biological high throughput screening systems,…) and methods to increase the throughput in the specific sub-processes (high throughput screening / high content screening / high throughput analytics,…) over the last few years [2]-[5]. However complex automation demands on wide-ranging process chains have often met their limits in integration. Isolated solutions for specialized operations such as synthesis and analytics still often dominate in lab automation. There are also complex automation systems, usually robot-assisted, that support more wide-ranging functionalities. Results include complex screening systems for biological testing of potential active agents. The systems available have often only covered a limited part of a continuous and interdependent workflow in lab applications up to now; processes such as synthesis, analysis and biological testing of new potential active agents require different systems that need to be adapted to the application requirements that apply.

Automated systems are usually controlled and monitored via a largely self-contained process control system (PCS) that provides users with specific control panels for process definition and control. Local process control by PCS for heterogeneous components in a robotic system and, increasingly, dynamic local optimization of procedure has become the state of the art [6]-[10]. Process modeling and notation as well as execution languages are only locally valid for certain robotic systems, as they are usually proprietary and manufacturer-dependent.

Several IT-solutions are established for specific parts of data management. These are used parallel for different activities: instrument-specific PCS capture data from analytical measurements (e. g. chromatography, mass spectrometry, optical methods like fluorescence or absorption spectroscopy et al.), bioinformatics systems deal with biological data (genome analysis, sequence analysis, structural bioinformatics), LIMS (Laboratory Information Management Systems) serve the sample, order, research study, and project management, CIMS (Chemical Information Management Systems) register chemical structures and their features, and DMS (Document Management Systems) manage operating procedures (SOP's) [11]. The standardized data exchange between these manifold systems together and with tools for data analysis, statistics, and visualization (like MS Excel, TIBCO Spotfire, MathWorks Matlab etc.) is still insufficiently. The handling of the exponentially growing of data and information flood is often a weakness in the laboratory workflow. Alternative algorithms and calculation rules with a low automation level are used for the evaluation of the data. Data driven decisions are often integrated. The results are available as specific files or reports or in databases. Personal undocumented knowledge reduces the traceability and the reproducibility. The comprehensive obligatory documentation requirement – in sub-processes is taken for granted - cannot be fulfilled [12]. For these reason the process of knowledge extraction in research projects is delayed.

1.2 Previous Works

The authors have been working successfully in several innovation levels of information systems over the past years. One of the first fully web based laboratory information management system (LIMS) with online automation networking was developed and was successfully integrated in a research center of the chemical industry [13] in addition to the use in the center for life science automation (celisca). This LIMS is application-independent by a user-defined parameter library and a method repository. Currently chemical, biological, and preventive medical research groups work with this system. Core functionalities are the administration, the organization and documentation of the laboratory workflows and research projects on user-defined abstraction level and with any structuring. An electronically laboratory notebook (ELN) is integrated for the unstructured process documentation in the development as well as operation phase. The progressive technical automation inclusive the generated data amounts requires an automation of data management, data manipulation and knowledge extraction. Therefore the basic LIMS are expanded from a pure measurement management system to an integration platform. This is a framework solution for a user-defined process description, a configurable process communication for the automated exchange of process data, the process visualization as well as a template based process data manipulation [14],[15]. Additional parts support the compound library management and the document management. In summary it is a data warehouse for all relevant data and information in life sciences applications (raw data, related meta data, derived data, process descriptions, information about chemical structures, storage information,…) expanded by facilities for data manipulation, visualization and communication. Using such system is a basic necessity without that a data interpretation and knowledge extraction is not effective practicable by reason of the enormous data amount in the environment of high throughput screening and high content applications (HTS/HCS).

The limits of this system are in consequence of its interactivity. For example, the detail level of the process description is user-defined. The user has many possibilities to integrate freestyle text, graphics, photos, documents. But it is the decision of the users, in which degree they use this possibilities in a specific process description. Because of this the user has the responsibility for the completeness and the traceability of a process description. However a complete process documentation is very important for the quality assurance and the quality assessment.

A networking of process descriptions is possible but the users apply this across working groups and across applications only rarely. The LIMS is an information and management system for measurements and experiment results. There are two ways of assignment to each other. The manual allocation by import or capture requires a high accuracy of the user to assign the measurements to the correct process instance. The automated way by the process communication uses for example the barcode identification. The configurable process communication allows the data exchange by files, databases and XML. These interfaces offer only the indirect way of device controlling with manual intervention due to a lack of integration facilities on side of devices. Connected to that, a timer-controlled collecting of data from lab instruments and an user-initiated visualization of this data provide only a limited monitoring.

2 BPM-Approach in LSA

Automation is established for the execution of the actual experiment. Preparatory and accompanying sub-processes (storage, maintenance, method development), are mostly time-consuming manual workflows with insufficient process integration. The use of methods, techniques and tools of BPM for agile processes in life sciences moves the overall process into the center of attention. The final goal is to develop an overall system with a comprehensive reproducible process control and process monitoring from the established autonomous automated sub systems (pipetting robot, multi parallel reactors,…) or islands of automation (robot lines that automate complex, high level sub-processes). In addition to these fully-automated sub-processes, manual process steps and activities, which are generating decisions and knowledge, will be integrated. The comprehensive automation of the overall workflow, which covers all involved, any structured automation components across the application fields is the logical further development of the automation of individual working steps. BPMS takes over the role of a systems integrator or rather the role of an integration platform for the process control and support so the interoperability. Means and methods of the library-based process definition and process documentation are needed.

In the interdisciplinary area of life science automation several disciplines cooperate in a common project. It is a major advantage to speak the same language. A documented process description in BPMN provides a uniform basis for the implementation of an automated process and its execution.

The direct controlling of laboratory instruments and a central resource management are realizable based on a service-oriented architecture (SOA). The overall process controlling reduces the down times. Further advantages are accrued for the quality assurance: the kind and the form of the documentation are defined once and are applied for all instances in the same manner. Therefore the reproducibility is guaranteed: there is a detailed process description in form of an executable model and the personal knowledge is integrated in this model. The integration of data analysis tools reduces the need for time-consuming and cost-intensive manual intervention.

2.1 Requirements on BPM in LSA

For an automation of the overall process are required:

- Integration of laboratory instruments by several technical interfaces
- Agile R&D processes with short life cycles require a high flexibility; there are unstructured sub-processes
- The process models are understandable for the process experts and adequate detailed for an process control
- Common repository for process models allows a reusing of sub-process models by adaption and parameterization
- Support of the business process life cycle: process analysis, modeling, execution with resources scheduling (instruments, lab personnel, services, software), monitoring, optimization based on analysis of results

- Integration of the human workflow (user task management) and external partners (e.g. e-mail or SMS to initiate manual intervention in error case or as status information)
- Data-driven decisions
- Rule management (easy to edit business rules, that control process activities, allow a fast adaption of the process und support data-driven decisions)
- Integration of several software products (data acquisition, data analysis, visualization)
- Data manipulation across the instances (common evaluation of experimental series)
- Integration of the document management including document generation, editing and exchange integrated in the message flow
- Support of quality assurance.

2.2 The Concept of BPM Integration

The approach of figure 1 shows an open BPMS, which communicates with any heterogeneous and hybrid components. The displayed SOA-components cover reprehensive important IT-solutions of the hierarchical laboratory automation. A set of adapter and connectors to the several systems is needed to control and exchange data. Sub-processes with integrated connectors, which are stored in the repositories, can easy be combined in new process models by the expert staff. Personal knowledge will be generally available. A BPMS integrates all stakeholder of the process. Thereby the advantages of the BPMS are opened up for the research processes, that are very flexible, unstructured, previously not completely defined, and often data driven. A major challenge is the handling of large data amounts integrated in the message flow of the overall process.

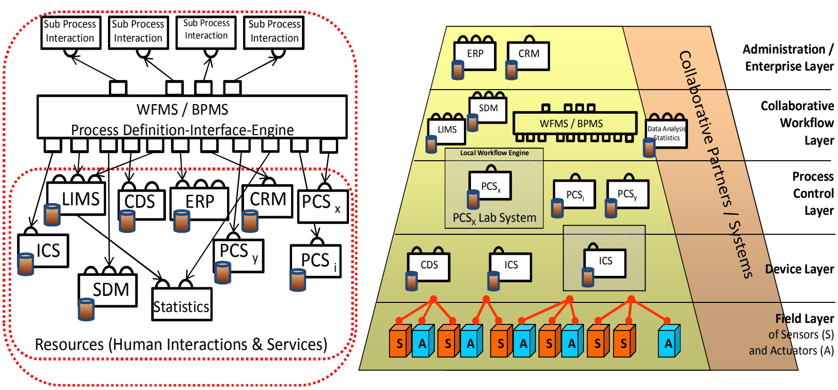


Fig. 1. General view of BPMS as integration platform in life science automation

(Laboratory information management systems - LIMS, Chromatography Data System – CDS, Instrument Control System - ICS, Scientific Data Management – SDM, Process Control Systems – PCS, Enterprise Resource Planning – ERP, Customer Relationship Management - CRM)

2.3 First Results and Future Works - An Example R&D Process

The in the following described models are developed in an evaluation study that investigates the applicability of BPM methods in the life science automation. Workflows of chemical, biological and medical applications are analyzed within this study.

First results are related to a project of the preventive medicine. This following example (Figure 2) describes a telematics platform solution for the fitness and stress monitoring applications based on mobile data acquisition with wireless personal area networks [16]. This system is used in preventive medicine studies, that investigate for example the consequences of the automation for the laboratory staff.

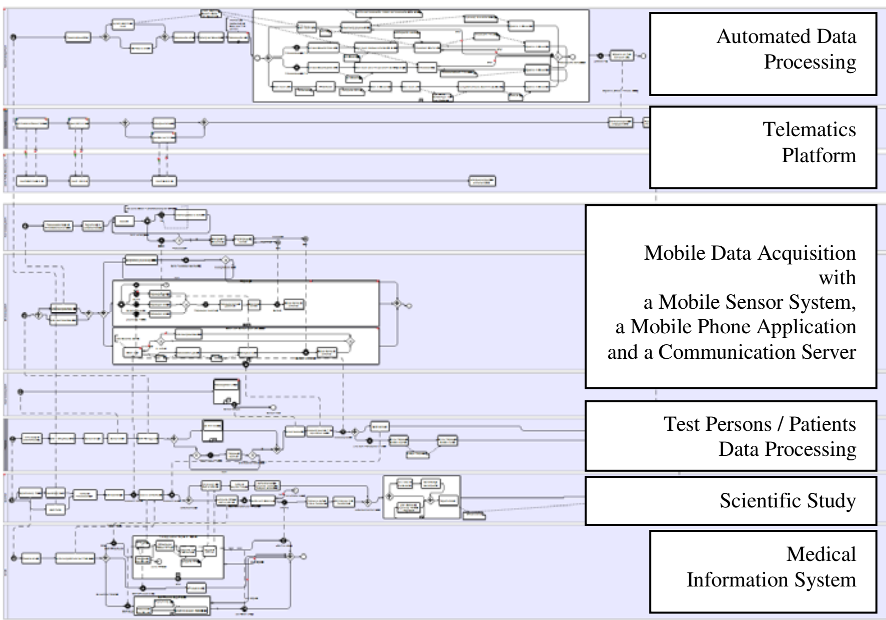


Fig. 2. Model of the overall process of the example telematics application for stress and fitness monitoring

The patient is continuously monitored in the every-day-activities. Patients use their usual mobile phone and a wireless sensor system for the acquisition of several physiological parameters. In addition, test persons are able to add parameters describing subjective assessments during the executed activities, e.g., to document their current physical and psychical state to get an according subjective strain rating. All these data are transferred by wireless communication to a communication server that stores this data in a raw database [17]. Fully-automated, intelligent data processing modules derive secondary information about stress and fitness in in-time calculations (Figure 3). As result for the patient a graphical classification of the calculated value is available on a web-based portal. In a parallel research study the captured and derived data is investigated relating to influences on the physiological parameters, and develop algorithms and models for the calculation of further secondary information.

For the data management within this research study the medical information system (1.2) are used. A first part of this workflow is shown in Figure 4. Manual and user tasks dominate this sub-process now. An increased automation level is preferable.

For the modeling of this processes BPMN are used with the goal to execute. With BPMN 2.0, there is a standard that provides a modeling notation as well as a process execution language.

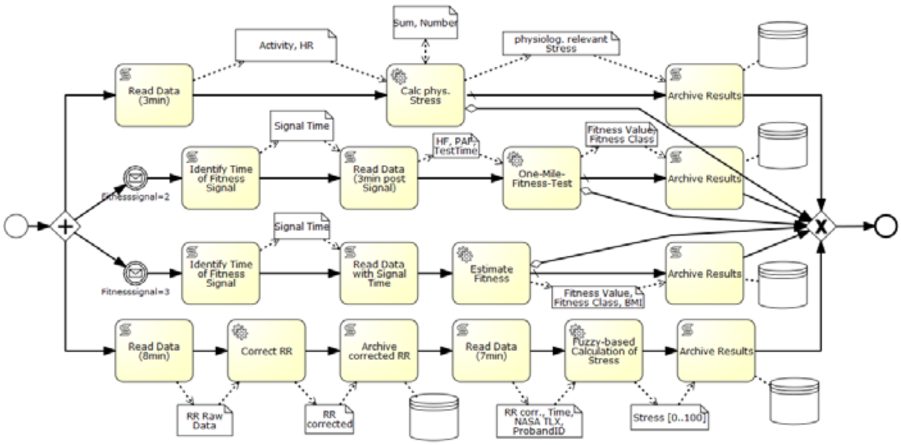


Fig. 3. Model of the sub-process of data processing executed by web services

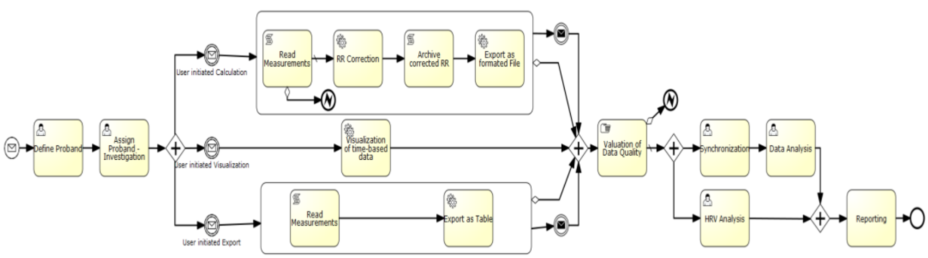


Fig. 4. Model of a part of the scientific study using the information system

In a first step of the study several BPMS are evaluated. Two basically different ideologies are to cite: BPMS based on zero-code-programming and developer-friendly BPMS. For the evaluation study at first the community version of intalio|BPMS [18] and the open source collaboration platform Activiti [19] are used. The modeling with the intalio Designer is intuitive. Simple processes with user task (a dialog editor is integrated), database access, integration of web services and sending of e-mails are implemented comfortable by process experts with few IT skills (SQL and XML is useful). Data structures can be defined and filled with content by mapping. The current version of intalio|BPMS does not support BPMN 2.0 specification. Our investigations will be continued, if intalio|Cloud is ready to use [18]. Activiti is a BPM – platform targeted at business people, developers, and system admins. It is open-source and distributed under the Apache license. Activiti offers many possibilities for use: the engine runs in any Java application, on a server or in the cloud. The tooling around Activiti facilitates an innovative and practical collaboration between business people and developers. Its core is a BPMN 2.0 process engine that has the focus on being light weight and easy to use for Java developers. The components of Activiti support the process modeling, the human task management, the administration, the running, the monitoring, and the collaboration between business people, developers and IT operational people. Thus Activiti is a complete BPMS but also a framework for developer, with that an own application can be realized [19]. In the future works, further life science applications are realized with the focus on scheduling of lab instruments and lab robots, aspects of quality assurance, and knowledge-intensive sub-processes.

Acknowledgments. The German Research Association (DFG) is gratefully acknowledged for funding this project.

References

1. Kalorama Information: The Worldwide Market for Lab automation. New York, USA (2008)
2. Elands, J.: The Evolution of Laboratory Automation. In: Seethala, R., Fernandes, P.B. (eds.) Handbook of Drug Screening, pp. 477–492. Marcel Dekker, New York (2001)
3. Haney, S.A. (ed.): High Content Screening – Science, Techniques and Applications. Wiley-Interscience Series on Mass Spectrometry. Wiley Interscience, Hoboken (2008)
4. Janzen, W.P.: High Throughput Screening - Methods and Protocols. Humana Press Inc., Totowa (2002)
5. Zhang, M., Nelson, B., Felder, R. (eds.): Life Science Automation: Fundamentals and Applications. Artech House Inc., Boston (2007)
6. Moore, K.W., Newman, R., Chan, G.K.Y., Leech, C., Allison, K., Coulson, J., Simpson, P.B.: Implementation of a High Specification Dual-Arm Robotic Platform to Meet Flexible Screening Needs. JALA - Journal of the Association for Laboratory Automation 12(2), 115–123 (2007)
7. Russo, M.F., Sasso, A.: Modeling, analysis, simulation and control of laboratory automation systems using Petri nets: Part 1- Modeling. JALA - Journal of the Association for Laboratory Automation 10(3), 172–181 (2005)
8. Bühl, W., Daniel, J., Höynck, M., Jänicke, W.B., Szarowski, S.: Laboratory resource planning for quality control of pharmaceuticals. Pharmazeutische Industrie 66(11 A), 1430–1434 (2004)
9. Rodziewicz, P., Bell, B.: Overview and architecture of the Java integration framework, hybrid scheduler, and Web-enabled LIMS. JALA - Journal of the Association for Laboratory Automation 9(6), 411–420 (2004)
10. King, J., Gosine, R., Delaney, B., Norvell, T., O'Young, S.: Discrete event control and dynamic scheduling for tele-robotic mining. CIM Bulletin 96(1069), 116–118 (2003)
11. Lämmel, J.: Systemintegration im Bereich der Laborinformatik. LaborPraxis 10 (2004)
12. Dellert-Ritter, M.: LIMS - Aktueller Stand und zukünftige Trends. GIT Laborzeitschrift 54(1), 39–42 (2010)

13. Thurow, K., Göde, B., Dingerdissen, U., Stoll, N.: Laboratory Information Management Systems for Life Science Applications. Organic Process Research & Development 8(6), 970–982 (2004)
14. Göde, B., Holzmüller-Laue, S., Haller, D., Schneider, I., Thurow, K.: Flexible IT-Plattform zur automatisierten HTS-Wirkstoffanalyse. GIT 9, 741–744 (2007)
15. Göde, B., Holzmüller-Laue, S., Rimane, K., Thurow, K.: Integrierte flexible Datenverarbeitung in einem webbasierten LIMS: Idee und Praxis eines Excel-Prozessors in Serverapplikationen. Chemie – Ingenieur – Technik 12, 2043–2049 (2007)
16. Berndt, R.D., Takenga, M.C., Kuehn, S., Preik, P., Stoll, N., Thurow, K., Kumar, M., Weippert, M., Rieger, A., Stoll, R.: A Scalable and Secure Telematics Platform for the Hosting of Telemedical Applications. In: Case Study of a Stress and Fitness Monitoring. IEEE, 13th International Conference on e-Health Networking, Application and Services, Columbia, USA, June 13-15 (accepted, 2011)
17. Neubert, S., Arndt, D., Thurow, K., Stoll, R.: Mobile Real-Time Data Acquisition System for Application in Preventive Medicine. Telemedicine and e-Health 16(4), 504–509 (2010)
18. http://www.intalio.com
19. http://www.activiti.org

A Conceptual Framework for Design Science Research

Łukasz Ostrowski, Markus Helfert, and Fakir Hossain

School of Computing, Dublin City University, Glasnevin, Dublin 9, Ireland
{Lostrowski,Markus.Helfert,Fakir.Hossain}@computing.dcu.ie

Abstract. Based on reviewing foremost literature, the paper discusses design science research methodology with strategies for design science evaluation. We have identified an interesting problem that, for certain artefacts, this methodology leads to a recursion effect. Therefore, we propose a conceptual framework for these artefacts, which, once developed, will detail facets of design science methodology for producing abstract design knowledge. The framework aims to fill out the lack of details in design science methodology for development of abstract design knowledge.

Keywords: Design Science Research Methodology, Design Science Evaluation, Abstract Design Knowledge.

1 Introduction

Over the last years, design science research has received increased attention in computing and Information Systems (IS) research. Generally, research, as a process, "is the application of scientific method to the complex task of discovering answers (solutions) to questions (problems)" [1]. We can differentiate between the study of natural systems such as physics, biology, economics and sociology [2], and the creation of artificial ones such as medicine and engineering [2,3].The core mission of the former is to develop valid knowledge to understand the natural or social world, or to describe, explain and possibly predict. The centre of the latter is to develop knowledge that can be used by professionals in the field in question to design solutions to their field problems. Understanding the nature and causes of problems can be a great help in designing solutions. This is the focus of design science [4]. However, design science does not limit itself to the understanding, but also aims to develop knowledge on the advantages and disadvantages of alternative solutions [4]. The main requirements are rigour and relevance [5,6,7].

Although there have been numerous contributions related to design science, there are still some fundamental challenges related to this research process. In this paper we aim to demonstrate these challenges, in particular related to detailing design activities which produce abstract design knowledge. Those challenges form a conceptual framework that is based on related literature.

The remainder of the paper is organized as follows. First, we review the general process of design science research methodology. Next, we distinguish between abstract and situational design knowledge as different outcomes of design artefacts. Then, we discuss the multi-grounded theory along with its application in evaluation strategies for

J. Grabis and M. Kirikova (Eds.): BIR 2011, LNBIP 90, pp. 345–354, 2011.

design science research. Finally we present the conceptual framework that is used to develop an artefact that will detail design science methodology.

2 Design Science Research

A number of researchers, both in and outside of the Information Systems discipline, have sought to provide some guidelines to define design science research [6]. Their work in engineering [8,9,10,11], computer science [12,13], and IS [14,15,12,16,17,18,19] have aimed to collect and distribute the appropriate reference literature [20]; characterize its purposes, differentiate it from theory building and testing research and from other research paradigms.

They enhanced its essential elements; and claim its legitimacy. However, despite several guidelines the literature has not explicitly focused on development of a detailed methodology for carrying out design science research [21].

Some researchers in IS and other disciplines have contributed ideas for process elements [8,13,9,18,17,12]. These papers include some component in the initial stages of research to define a research problem. Nunamaker et al. [1] and Walls et al. [18] emphasized theoretical bases, whereas engineering researchers [8,9] focused more on applied problems. Takeda et al. [13] suggested the need for problem enumeration, whereas Rossi and Sein [17] advocated need identification. Hevner et al. [6] asserted that design science research should address important and relevant problems. Based on those representative papers, which stated or suggested process elements, the components of the Design Science Research Methodology (DSRM) were synthesized [21]. The result of the synthesis was a process model consisting of six activities in a nominal sequence. We describe them here and graphically in Figure 1.

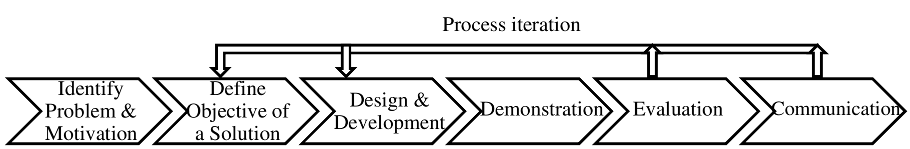


Fig. 1. DSRM Process Model [21]

Problem identification and motivation defines the specific research problem and justifies the value of a solution. Because the problem definition will be used to develop an artefact that can effectively provide a solution, it may be useful to atomize the problem conceptually so that the solution can capture its complexity [21].*Define the objectives for a solution* refers to the objectives of a solution from the problem definition and knowledge of what is possible and feasible. The objectives can be quantitative, such as terms in which a desirable solution would be better than current ones, or qualitative, such as a description of how a new artefact is expected to support solutions to problems not hitherto addressed [21]. *Design and development* creates artefacts. Such artefacts are potentially constructs, models, methods, or instantiations [6] or “new properties of technical, social, and/or informational resources” [3]. Conceptually, a design research artefact can be any designed object in which a research contribution is embedded in the design [6]. *Demonstration* refers to the use of the artefact to solve one or more instances of the problem. This could involve its use in experimentation, simulation, case study, proof, or other appropriate activity. Resources required for the demonstration include effective knowledge of how to use the artefact to solve the problem [21]. *Evaluation* observes and measures how well the artefact supports a solution to the problem. This activity involves comparing the objectives of a solution to actual observed results from use of the artefact in the demonstration. It requires knowledge of relevant metrics and analysis techniques [21]. *Communicate* refers to the problem and its importance, the artefact, its utility and novelty, the rigor of its design, and its effectiveness to researchers and other relevant audiences such as practicing professionals. This could be in form of a PhD thesis, journal or conference article [22]. In scholarly research publications, researchers might use the structure of this process to structure the paper, just as the nominal structure of an empirical research process (problem definition, literature review, hypothesis development, data collection, analysis, results, discussion, and conclusion) is a common structure for empirical research papers [21].

2.1 Outcome of Design Science: Abstract and Situational Design

Design research produces different artefacts with Design and Development activities of a DSRM process model [21]. The outcome of design science is an artefact, which can be in form of a construct, model, method, and an instantiation [16,6]. Some researchers understand artefacts as "things", i.e. entities that have some separate existence [23]. Constructs are defined as "concepts" and "conceptualizations" [16] and "vocabulary and symbols" [6]. These constructs are abstracted concepts aimed for theorizing and trans-situational use. "Conceptualizations are extremely important in both natural and design science. They define the terms used when describing and thinking about tasks" [16]. Models are not conceived as abstract entities in the same way as constructs. "Models use constructs to represent a real world situation – the design problem and its solution space…" [6] "Models aid problem and solution understanding and frequently represent the connection between problem and solution components enabling exploration of the effects of design decisions and changes in the real world." [6]. A method is defined as "a set of steps (an algorithm or guideline) to perform a task" [16]. An instantiation is a prototype or a specific working system or some kind of tool [23].

Design research uses and produces design knowledge, which can be either in a form of abstract or situational design knowledge [24]. Following this differentiation, Goldkuhl and Lind [24] divided design research into two activity layers: 1) design practice that produces situational design knowledge and concrete artefacts and 2) meta-design that produces abstract design knowledge. In this paper, we focus on abstract design knowledge produced in the meta-design activity. This can be viewed as 1) a preparatory activity before situational design is started and 2) a continual activity partially integrated with the design practice 3) a concluding theoretical activity summarizing, evaluating and abstracting results directed for target groups outside the studied design and use practices [24]. As a result we receive four different outcomes for each activity layer (see Table 1).

Table 1. Different outcomes differentiated into abstract vs. situational [24]

Activity type Outcome	From meta-design: Abstract design knowledge	From design practice: Situational design knowledge and results
Constructs	abstract concepts	situational concepts (may be applied and adapted from abstract concepts)
Models	generic models	situational models
Methods	guidelines for design practice	parts of a situational system or process
Instantiation	(systems abstractions with key properties)	Instantiations IT systems (prototype or working system)

2.2 Artificial and Naturalistic Evaluation

As discussed above, abstract and situational design knowledge can be treated as two individual outcomes of design science as stated in section 2.1. Thus, it seems reasonable to consider two different evaluation methods for each of them – artificial and naturalistic. Evaluation has been a topic both in general IS research and in design science research. In the general IS literature, evaluation is generally regarded from one of two perspectives [25]. In the *ex-ante* perspective, candidate systems or technologies are evaluated before they are chosen and acquired or implemented. In the *ex post* perspective, a chosen system or technology is evaluated after it is acquired or implemented [25].

Venable [26] classified design science research evaluation approaches into two primary forms: artificial and naturalistic evaluation. Artificial evaluation evaluates a solution technology in a contrived and non-realistic way. Naturalistic evaluation explores the performance of a solution technology its real environment (i.e. within the organisation). Naturalistic evaluation methods offer the possibility to evaluate the real artefact in use by real users solving real problems [27], while artificial evaluation methods offer the possibility to control potential confusing variables more carefully and prove or disprove design hypotheses, design theories, and the utility of design artefacts. Having taken into account those two dimensions, Pries-Heje et al. [28] introduced an evaluation framework. We applied the framework to the DSRM process model in Figure 2. We split the Design and Development activity into meta-design and design practice. Since different artefacts are achieved from each activity, a different evaluation strategy applies.

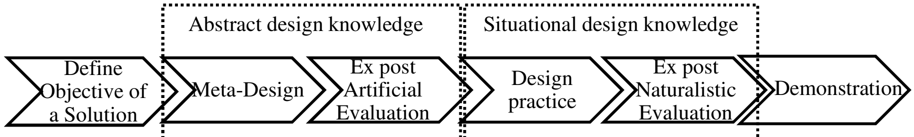


Fig. 2. A fragment of DSRM [21] model process with implemented strategic evaluation and different outcomes

Artificial evaluation is selected after the meta-design activity because of its capability to test design hypotheses [18]. Critical techniques may be used, but these generally supplement the main goal of proving or disproving the design theory and/or the utility of the design science artefact. Artificial evaluation includes laboratory experiments, field experiments, simulations, criteria-based analysis, theoretical arguments, and mathematical proofs [28]. Artificial evaluation is then unreal in some way or ways according to the three realities [27], such as unreal users, unreal systems, and especially unreal problems (not held by the users and/or not real tasks, etc.). The naturalistic evaluation was placed after the design practice outcome because of the capability of performing evaluation in a real environment (real people, real systems (artefacts), and real settings [27], naturalistic evaluation embraces all of the complexities of human practice in real organisations. Naturalistic evaluation methods include case studies, field studies, surveys, ethnography, phenomenology, hermeneutic methods, and action research. To the extent that naturalistic evaluation is affected by confusing variables or misinterpretation, evaluation results may not be precise or even truthful about an artefact's utility or efficacy in real use [28].

3 A Conceptual Framework for Meta-design

Drawing upon the above literature and our discussion we described design science methodology illustrated in Figure 3 (without callouts). Now, we will focus on the meta-design step which aims to design a desired artefact. Offerman [22] claimed that not much guidelines is provided in IS literature on this step, and proposed that focus should be put on relevant scientific publications (i.e. literature review) regarding the desired artefact. In his proposed design science research process, an expert survey and case study research activities are only involved in the evaluation step. We propose that for meta-design step, these activities along with expert's knowledge should be involved in designing artefacts, and treated as a separate entity (i.e. engagement scholarship) [29]. Since a literature review and engagement scholarship provide information regarding the desired artefact, a way of presenting relevant findings in a rigour structure seems reasonable. Thus, we propose, that modelling techniques should be accompanied to the mentioned sources of information. As a result we can distinguish three sub-steps involved in the meta-design step: literature review, modelling and engagement scholarship (see Figure 3).

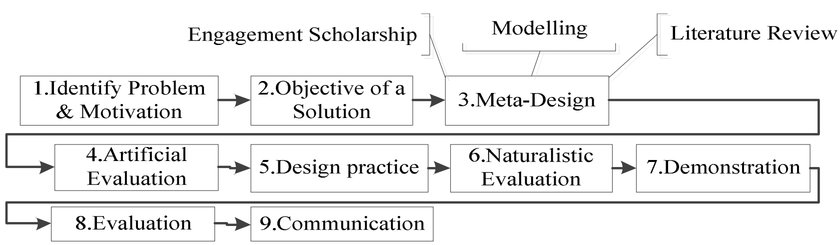


Fig. 3. Steps in design science methodology

Literature to date does not provide much detail on how to approach these three sub-steps in design science. Therefore, we propose the conceptual framework to develop an artefact that will detail the meta-design activity that produces abstract design knowledge. This is one of the challenges in the design science research process. One view is that, these three sub-steps might be seen as an individual artefact, so we could just reach for design science, because it aims to develop knowledge that can be used by professionals in the field in question to design solutions to their field problems [4]. However, this approach leads to an occurrence of a recursion effect, which will be shown in an example.

First, we will present the conceptual framework and its outcome, and then the example which underlines the fact that these three sub-steps can only be partly produced using design science methodology. It demonstrates how our conceptual framework deals with this issue. The application of the framework should result in detailing the meta-design step by providing guideline on how to approach literature review, engagement scholarship and modelling in design science methodology.

3.1 Details of the Conceptual Framework

In previous section we discussed that the three sub-steps (literature, review, engagement scholarship, modelling) in meta-design work together to produce abstract design knowledge. Figure 4 illustrates their dependence and supports our claim that these three sub-steps cannot be approach individually. In other words we need to consider the three activities as a one artefact.

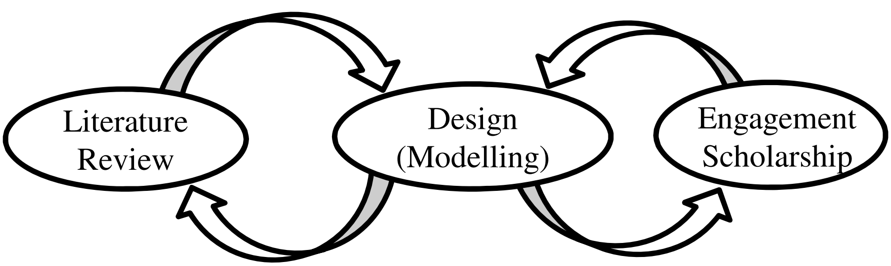


Fig. 4. Activities in meta-design step

Therefore we approach them simultaneously and refer to the definition of IS which is the applied research discipline [21]. In the sense, that we frequently apply theory from other disciplines, such as economics, computer science, and the social sciences, to solve problems at the intersection of information technology and organizations [21]. Thus, the literature review activity and two others can be produced by comparing multiple plausible models of reality, which are essential for developing reliable scientific knowledge [30]. However, these models would still need redesigning to fit the purpose of the output in meta-design step. This is why we need the conceptual framework. The idea is to apply theories of disciplines of the three sub-steps to the outcome level of the artefact of interest, and then combine. For example, we intuitively reach for plausible constructs in domains of literature review, engagement scholarship, and modelling. Then we come up with a common construct for each domain, and combine, so that in a result we get a general construct on how to conduct these three sub-steps all together (Figure 5) in meta-design step. Upon having the construct we go deeper into these domains to accordingly get model, method, and instantiation. Following this approach we will produce an artefact that provides substantial guidelines on how to conduct literature review, engagement scholarship, and modelling in meta-design step for the purpose of an abstract design knowledge artefact. In other words, by using this framework we will detail the meta-design step in design science.

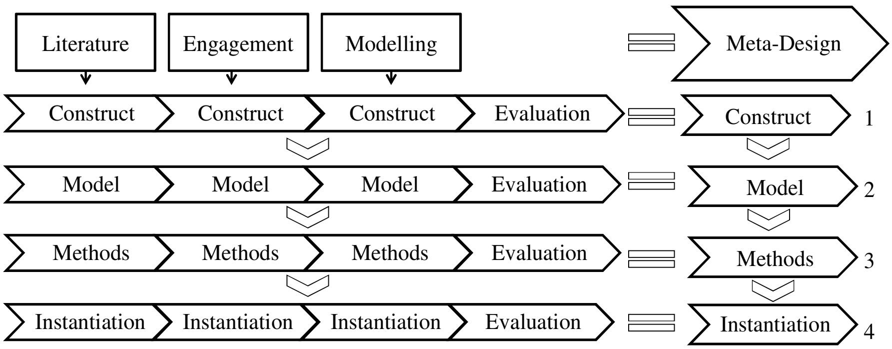


Fig. 5. A conceptual framework for meta-design artefact

The word *evaluation* also appears in our framework. To keep it clear, the artificial evaluation was a part of the abstract design knowledge in our earlier discussion (Figure 2 and 3). The difference between it and the evaluation, used in the conceptual framework (Figure 5), is that the evaluation is used in the process of producing the artefact of three sub-steps for meta-design as oppose to evaluating the artefact, which is done by the artificial evaluation. Our new-created meta-design artefact will be the guideline for the three sub-steps showed in Figure 4. We believe that the embedded evaluation will cause lower incorrectness of the guideline upon applying artificial evaluation.

3.2 Example: Design Science for Producing a Literature Review Artefact

In this example, we use design science methodology discussed in previous sections, and presented in Figure 3. Accordingly the first step is to *identify a problem & motivation*. In our case, we found that a systematic literature review would increase a process of retrieving relevant information about a domain under consideration. We also noticed that IS researchers do not provide enough details on the literature review activity in design science methodology. We are motivated to specify it, and distribute to others. Next step in design science is to *define objective of a solution*. We aim to provide a systematic guideline on literature review that would help other researchers thoroughly conduct this activity. The solution will be in form of rules and steps that need to be fulfilled to reach desired outcome. The following step is *meta-design* in which we produce the abstract design knowledge. Four possible outcomes are available (see Table1). We understand that the instantiation for meta-design should base on the method, which were embedded in the model whose beginning can be found in the construct. In the other words, the instantiation is a cumulative outcome at some stage (see Figure 5 for the linear approach). Thus, to achieve an instantiation of the literature review artefact, we need to produce a construct first. According to our discussion summarized in Figure 3, we use sub-steps such as engagement scholarship, modelling and literature review to produce the construct for desired literature review artefact. Consequently, we reach first for the literature review sub-step to gain information needed to develop the construct. This sub-step should tell us what actions we are required to take; however, since we are designing the literature review artefact, we do not know yet how this activity should look like. Hence, we could follow the line of thought and ask the question on how to conduct the literature review activity. If we reached for design science again to design the artefact of literature review, we would end up at the same step. Thus, this reasoning leads us to the recursion effect. By way of explanation we need to know the literature review in order to design the literature review. As a result the design science methodology for the literature review artefact does not seem appropriate. Otherwise, we would raise the same question over and over again once the sub-step in meta-design is reached. We get the same conclusions upon using design science methodology for the rest two sub-steps: engagement, and modelling if we aim to produce modelling artefact for example. This recursion effect is caused by two factors, first is that, as we stated in previous section, these three sub-steps are related to each other (Figure 4). We need to model results upon conducting a literature review and engagement scholarship. Therefore these steps should not be considered individually. The second is that you cannot use methodology to produce an artefact, which appears as a method in that methodology. In order to overcome the recursion effect we came up with a solution that requires designing these sub-steps in parallel. The solution to that solution is our conceptual framework and the artefact, developed thanks to it, will detail meta-design step in design science methodology.

4 Conclusions

In this paper we presented how design science research methodology (DSRM) [21] corresponds with the strategies for design science research evaluation [28] (Figure 3). We claimed, following Offerman's line of thought [22], that these three sub-steps (literature review, engagement scholarship, modelling) are crucial in meta-design step in design science. We indicated that this methodology can only be partly applied to produce artefacts for those three sub-steps. By doing so, we ended up in an endless loop. That also showed that DSRM is not detail enough to provide researchers with systematic guidelines. Thus we came up with a conceptual framework, which required reaching for disciplines of literature review, engagement scholarship, and modelling. These disciplines are used and combined in parallel to achieve a desired outcome, which is to detail the sub-steps for meta-design in design science methodology; provide researchers with a systematic approach while producing output in meta-design step. The outcome of the conceptual model will be applied and validated via case studies in our future research.

References

1. Aker, J.F., Chen, M., Purdin, T.D.M.: Systems Development in Information Systems Research. Journal of Management IS 7(3), 89–106 (1991)
2. Van Aken, J.: Management Research Based on the Paradigm of the Design Sciences: the Quest for Tested and Grounded Technological Rules. Journal of Management Studies 41(2), 219–246 (2004)
3. Simon, H.: The Sciences of the Artificial, 1st edn. MIT Press, Cambridge (1970)
4. Van Aken, J.E.: Management Research as a Design Science: Articulating the Research Products of Mode 2 Knowledge Production in Management. British Journal of Management 16(1), 19–36 (2005)
5. Benbasat, I., Zmud, R.W.: Empirical Research in Information Systems- The Practice of Relevance. MIS Quarterly 23(1), 3–36 (1999)
6. Hevner, A.R., March, S.T., Park, J., Ram, S.: Design Science in Information Systems Research. MIS Quarterly 28, 75–106 (2004)
7. Rosemann, M., Vessey, I.: Toward Improving the Relevance of Information Systems Research to Practice: The Role of Applicability Checks. MIS Quarterly 32(1), 1–22 (2008)
8. Archer, L.B.: Systematic Method for Designers. In: Developments in Design Methodology, London, pp. 57–82 (1984)
9. Eekels, J., Roozenburg, N.F.M.: A Methodological Comparison of the Structures of Scientific Research and Engineering Design-Their Similarities and Differences. Design Studies 12(4), 197–203 (1991)
10. Fulcher, A.J., Hills, P.: Towards a Strategic Framework for Design Research. Journal of Engineering Design 7(1), 183–193 (1996)
11. Reich, Y.: The Study of Design Methodology. Journal of Mechanical Design 117(2), 211–214 (1994)
12. Preston, M., Mehandjiev, N.: A Framework for Classifying Intelligent Design Theories. In: Mehandjiev, N., Brereton, B. (eds.) The 2004 ACM Workshop on Interdisciplinary Software Engineering Research, New York, pp. 49–54 (2004)
13. Takeda, H., Veerkamp, P., Tomiyama, T., Yoshikawam, H.: Modelling Design Processes. AI Magazine 11(4), 37–48 (1990)
14. Adams, L., Courtney, J.: Achieving Relevance in IS Research via the DAGS Framework. In: 37th Annual Hawaii International Conference on System Sciences (2004)
15. Cole, R., Purao, S., Rossi, M., Sein, M.K.: Being proactive- Where Action Research Meets Design Research. In: 26th International Conference on Information Systems, Atlanta, pp. 325–336 (2005)
16. March, S., Smith, G.: Design and Natural Science Research on Information Technology. Decision Support Systems 15(4), 251–266 (1995)
17. Rossi, M., Sein, M.K.: Design Research Workshop: A Proactive Research Approach. In: 26th Information Systems Research Seminar in Scandinavia, Haikko, pp. 9–12 (2003)
18. Walls, J., Widmeyer, G., El Sawy, O.: Building an Information System Design Theory for Vigilant EIS. Information Systems Research 3(1), 36–59 (1992)
19. Venable, J.R., Travis, J.: Using a Group Support System for the Distributed Application of Soft Systems Methodology. In: Hope, B., Yoong, P. (eds.) 10th Australasian Conference on Information Systems, Wellington, New Zealand, pp. 1105–1117 (1999)
20. Vaishnavi, V., Kuechler, B.: Design Research in Information Systems. Association for Information Systems (2005)
21. Peffers, K., Tuunanen, T., Rothenberger, M.: A Design Science Research Methodology. Journal of Management Information Systems 24(3), 45–77 (2007)

22. Offermann, P., Levina, O., Schonherr, M., Bub, U.: Outline of a Design Science Research Process. In: Design Science Research in Information Systems and Technology (2009)
23. Goldkuhl, G.: Design Theories in Information Systems – A Need for Multi-Grounding. Journal of Information Technology and Application 6(2), 59–72 (2004)
24. Goldkuhl, G., Lind, M.: A Multi-Grounded Design Research Process. In: Winter, R., Zhao, J.L., Aier, S. (eds.) DESRIST 2010. LNCS, vol. 6105, pp. 45–60. Springer, Heidelberg (2010)
25. Klecun, E., Cornford, T.: A Critical Approach to Evaluation. European Journal of IS 14(3), 229–243 (2005)
26. Venable, J.: A Framework for Design Science Research Activities. In: The 2006 Information Resource Management Association Conference, Washington DC (2006)
27. Sun, Y., Kantor, P.: Cross-Evaluation: A New Model for Information System Evaluation. Journal of the American Society for Information Science and Technology 57(5), 614–662 (2006)
28. Pries-Heje, J., Baskerville, R., Venable, J.: Strategies for Design Science Research Evaluation. In: 16th European Conference on Information Systems, pp. 255–266 (2008)
29. Van de Ven, A.: Engaged Scholarship: A Guide for Organizational and Social Research. Oxford University Press, New York (2007)
30. Azevedo, J.: Mapping Reality: An Evolutionary Realist Methodology for the Natural and Social Sciences. Albany (1997)

Author Index

Abramowicz, Witold 231
Aseeva, Natalia 110

Babkin, Eduard 110
Bassano, Clara 304
Bauler, Pascal 83
Brandão, Patrícia Leite 254
Bukhsh, Faiza Allah 281
Buksa, Ilze 96
Būmans, Guntars 142
Busch, Marianne 239

Čerāns, Kārlis 142
Chmielarz, Witold 267

Feltz, Fernand 83
Fill, Hans-Georg 29

Galvin, Maureen 304
Göde, Bernd 336
Görlach, Katharina 321
Grabis, Jānis 217

Helfert, Markus 167, 345
Holzmüller-Laue, Silke 336
Hossain, Fakir 345

Indihar-Štemberger, Mojca 178

Kalibatiene, Diana 124
Karagiannis, Dimitris 19
Karastoyanova, Dimka 321
Karpunina, Margarita 110
Knapp, Alexander 239
Kobyliński, Andrzej 296
Koch, Nora 239
Kovačič, Andrej 178
Kozlovics, Sergejs 190
Kozmina, Natalija 44
Krogstie, John 1

Leymann, Frank 321
Lodde, Andreas 83

Malets, Polina 321
Małyszko, Jacek 231
Meyer, Martin 167
Michalski, Marcin 296
Mikulovs, Igors 217

Niedrite, Laila 44
Niedritis, Aivars 44
Niemi, Tapio 59
Niinimaki, Marko 59
Nummenmaa, Jyrki 59

O'Brien, Conor 167
Ostrowski, Łukasz 345

Paduano, Erica 304
Pajk, Dejan 178
Piciocchi, Paolo 304

Rikacovs, Sergejs 158
Rocha, Álvaro 254
Rudzajs, Peteris 96

Sandkuhl, Kurt 204
Schlechter, Antoine 83
Schumm, David 321
Sonntag, Mirko 321

Thanisch, Peter 59

Vasilecas, Olegas 124
Victor, Avelino 254
Visic, Niksa 19

Węckowski, Dawid Grzegorz 231
Weigand, Hans 281

Zuters, Janis 73