Witold Abramowicz (Ed.)

Business Information Systems

LNBIP 157

16th International Conference, BIS 2013
Poznań, Poland, June 2013
Proceedings

Springer

Lecture Notes
in Business Information Processing 157

Series Editors

Witold Abramowicz (Ed.)

Business Information Systems

16th International Conference, BIS 2013
Poznań, Poland, June 19-21, 2013
Proceedings

Springer

Volume Editor

Witold Abramowicz
Poznań University of Economics
Department of Information Systems
al. Niepodległości 10
61-875 Poznań, Poland
E-mail: w.abramowicz@kie.ue.poznan.pl

ISSN 1865-1348 e-ISSN 1865-1356
ISBN 978-3-642-38365-6 e-ISBN 978-3-642-38366-3
DOI 10.1007/978-3-642-38366-3
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2013938616

ACM Computing Classification (1998): J.1, H.4, H.3

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

The 16th edition of the International Conference on Business Information Systems was held in Poznań, Poland. From the very beginning, BIS was recognized as an event that brings together scientific community, people involved in the development of business computer applications, as well as consultants helping to properly implement computer technology and applications in the industry.

Business information systems are subject to rapid development and innovation. The BIS conference follows trends in academia and business research, and thus the theme of the conference was "Business Applications on the Move." Sales of mobile computing devices increase every year, and the popularity of mobile devices has significant implications for business applications. Therefore, classic topics like business processes or information management have to be rethought.

This trend was reflected in the papers presented at the BIS conference. Thus, the first session focused on modern enterprises and mobile ERP systems. Moreover, during the conference, up-to-date topics were discussed, e.g., linked data and recommendations systems. Issues raised during the the previous editions of BIS continued to be discussed this year. This included papers from the areas of business models, BPM, ontologies, knowledge discovery IT frameworks, and systems architecture.

The regular program was complemented by outstanding keynote speakers: Martin Bichler (Department of Informatics of TU München, Munich, Germany), Manfred Hauswirth (Digital Enterprise Research Institute (DERI), Galway, Ireland), and Günter Müller (Institute of Computer Science and Social Studies, Department of Telematics, Albert Ludwig University of Freiburg, Germany).

The Program Committee consisted of almost 100 members who carefully evaluated all the submitted papers. Based on their extensive reviews, a set of 18 papers were selected, grouped into six sessions.

June 2013 Witold Abramowicz

Conference Organization

BIS 2013 was organized by Poznań University of Economics, Department of Information Systems.

Local Organization

Elżbieta Bukowska (Chair)	Poznań University of Economics, Poland
Barbara Gołębiewska	Poznań University of Economics, Poland
Włodzimierz Lewoniewski	Poznań University of Economics, Poland
Szymon Łazaruk	Poznań University of Economics, Poland
Bartosz Perkowski	Poznań University of Economics, Poland
Wioletta Sokołowska	Poznań University of Economics, Poland
Milena Stróżyna	Poznań University of Economics, Poland
Krzysztof Węcel	Poznań University of Economics, Poland

Program Committee

Witold Abramowicz	Poznań Univeristy of Economics, Poland
Dimitris Apostolou	University of Piraeus, Greece
Christian Bizer	Freie Universität Berlin, Germany
Michelangelo Ceci	Università degli Studi di Bari, Italy
Wojciech Cellary	Poznań, University of Economics, Poland
Dickson K.W. Chiu	Dickson Computer Systems, Hong Kong, SAR China
Oscar Corcho	Universidad Politécnica de Madrid, Spain
Tommaso Di Noia	Politecnico di Bari, Italy
Ciprian Dobre	University Politehnica of Bucharest, Romania
Schahram Dustdar	Technical University of Vienna, Austria
Suzanne Embury	University of Manchester, UK
Vadim Ermolayev	Zaporozhye National University, Ukraine
Agata Filipowska	Poznań University of Economics, Poland
Vladimir Fomichov	Higher School of Economics, Russia
Bogdan Franczyk	University of Leipzig, Germany
Flavius Frasincar	Erasmus University Rotterdam, The Netherlands
Johann-Christoph Freytag	Humboldt Universitä zu Berlin, Germany
Marko Grobelnik	Jozef Stefan Institute, Slovenia
Norbert Gronau	University of Potsdam, Germany
Volker Gruhn	Universität Duisburg-Essen, Germany
Francesco Guerra	University of Modena, Italy
Jon Atle Gulla	Norwegian University of Science and Technology, Norway

Hele-Mai Haav	Tallinn University of Technology, Estonia
Manfred Hauswirth	National University of Ireland, Ireland
Martin Hepp	Bundeswehr University of Munich, Germany
Knut Hinkelmann	University of Applied Sciences Northwestern Switzerland, Switzerland
Marta Indulska	The University of Queensland, Australia
Marijn Janssen	Delft University of Technology, The Netherlands
Adam Jatowt	Kyoto University, Japan
Monika Kaczmarek	Poznań University of Economics, Poland
Tomasz Kaczmarek	Poznań University of Economics, Poland
Pawel Kalczynski	California State University, USA
Gary Klein	University of Colorado, USA
Ralf Klischewski	German University in Cairo, Egypt
Jacek Kopecky	The Open University, UK
Jacek Koronacki	Polish Academy of Sciences, Poland
Ryszard Kowalczyk	Swinburne University of Technology, Australia
Marek Kowalkiewicz	SAP Research, Australia
Helmut Krcmar	Technische Universität München, Germany
Dalia Kriksciuniene	Vilnius University, Lithuania
Sergei O. Kuznetsov	National Research University Higher School of Economics, Russia
Daniel Lemire	Université du Québec à Montréal, Canada
Maurizio Lenzerini	University of Rome "La Sapienza", Italy
Frank Leymann	University of Stuttgart, Germany
Peter Loos	University of Saarbrücken, Germany
Qiang Ma	Kyoto University, Japan
Maria Mach-Król	Katowice University of Economics, Poland
Leszek Maciaszek	Wrocław University of Economics, Poland
Yannis Manolopoulos	Aristotle University of Thessaloniki, Greece
Florian Matthes	Technische Universität München, Germany
Heinrich C. Mayr	Alpen-Adria-Universitä Klagenfurt, Austria
Nor Laila Md Noor	Universiti Teknologi MARA, Malaysia
Jan Mendling	Wirtschaftsuniversität Wien, Austria
Günter Müller	University of Freiburg, Germany
Andreas Oberweis	Universitä Karlsruhe, Germany
Mitsunori Ogihara	University of Miami, USA
Marcin Paprzycki	Polish Academy of Sciences, Poland
Eric Paquet	National Research Council, Canada
Klaus Pohl	University of Duisburg-Essen, Germany
Jaroslav Pokorny	Charles University, Czech Republic
Elke Pulvermueller	University of Osnabrück, Germany
Thurasamy Ramayah	Universiti Sains Malaysia, Malaysia
Gustavo Rossi	National University of La Plata, Argentina

Massimo Ruffolo	University of Calabria, Italy
Shazia Sadiq	The University of Queensland, Australia
Virgilijus Sakalauskas	Vilnius University, Lithuania
Sherif Sakr	The University of New South Wales, Australia
Demetrios Sampson	University of Piraeus, Greece
Kurt Sandkuhl	University of Rostock, Sweden
Juergen Sauer	University of Oldenburg, Germany
Ulf Seigerroth	Jönköping University, Sweden
Gheorghe Cosmin Silaghi	Babes-Bolyai University, Romania
Elmar J. Sinz	University of Bamberg, Germany
Kilian Stoffel	University of Neuchâtel, Switzerland
Darijus Strasunskas	Norwegian University of Science and Technology, Norway
Jerzy Surma	Warsaw School of Economics, Poland
Bernhard Thalheim	Christian Albrechts University of Kiel, Germany
Barbara Thoenssen	University of Applied Sciences Northwestern Switzerland, Switzerland
Robert Tolksdorf	Freie Universität Berlin, Germany
Herna Viktor	University of Ottawa, Canada
Krzysztof Wecel	Poznań University of Economics, Poland
Mathias Weske	University of Potsdam, Germany
Anna Wingkvist	Linnaeus Univeristy, Sweden
Andreas Wombacher	University of Twente, The Netherlands
Qi Yu	Rochester Institute of Technology, USA
Slawomir Zadrozny	Polish Academy of Sciences, Poland
John Zeleznikow	Victoria University, Australia
Jozef Zurada	University of Louisville, USA

Additional Reviewers

Macchia, Lucrezia
Monahov, Ivan
Nastic, Stefan
Norkus, Oliver
Nowak, Alexander
Pio, Gianvito
Roa Marin, Henry
Schiele, Gregor
Schneider, Alexander
von der Weth, Christian
Xavier Parreira, Josiane

Table of Contents

Evaluating Cloud Services Using Methods of Supplier Selection

Stefan Harnisch and Peter Buxmann

Technische Universität Darmstadt, Chair of Information Systems,
Hochschulstraße 1, 64289 Darmstadt, Germany
{harnisch,buxmann}@is.tu-darmstadt.de

Abstract. The recent establishment of cloud computing and its increasing importance for consumers have attracted new ways of service creation and provisioning over the internet. Research has dealt with the establishment of new cloud computing markets and new opportunities for collaboration in these emerging markets. We found that the necessary decisions are very much similar to decisions that have to be made in "classic" offline supply chains. Based on methods for supplier selection, we develop and illustrate a method for the evaluation and selection of cloud services. The fuzzy AHP approach we propose allows decision makers to account for uncertainty in a proven, well-structured way.

Keywords: Collaborative cloud markets, service evaluation, supplier selection, AHP, fuzzy logic.

1 Introduction

Recently, a great deal of cloud services and platforms has become available to the public. There are services suited for many tasks and there may be several services which are able to fulfill a certain task. In this context, of late there has been research around emergent software [19], describing systems or software applications that are capable to evolve beyond design-time which allows for new ways of collaboration and combination. [42] illustrate the vision of a global cloud marketplace, also enabling and enforcing the collaboration of different services. In this paper, we take the viewpoint of a cloud service provider or developer and explore possibilities for the evaluation and selection of services that could be used or combined with other, respectively own, services.

We found that the necessary decisions are very much similar to decisions that have to be made in "classic" offline supply chains. Which of the manifold available suppliers and products can serve the purpose, the product is needed for, best? A lot of research has been conducted in this area of supplier selection and we seek to analyze the state-of-the-art in this context and transfer methods to the context of collaborative cloud services. That way, we can profit from previous research and bring new ideas from valuable existing studies to the area of business information systems. Therefore, we aim at answering the following research questions in this paper:

- Which of the existing methods for supplier selection is suited best for an application in the context of cloud computing?

W. Abramowicz (Ed.): BIS 2013, LNBIP 157, pp. 1–13, 2013.

– How can the existing, suitable methods be transferred for the usage in a cloud computing context?

After the identification of the state-of-the-art of methods of supplier selection we describe a fuzzy analytical hierarchy process (AHP) model based on the two most prominent methods for supplier selection. We design a first and new approach for its application to collaborative cloud services.

The remainder of the paper is structured as follows: In section 2, we summarize existing research and the state of the art concerning methods of supplier selection. We then evaluate the methods for our purpose. Section 3 deals with the conceptualization and illustration of our new approach. Finally, we conclude the paper in section 4, where we also mention the limitations of our work and present avenues for future research.

2 Application of Methods for Supplier Selection in the Context of Cloud Computing

The recent establishment of cloud computing and its increasing importance for consumers [4] has attracted new ways of service creation and provisioning over the internet (cf. e.g. [18]). Research has dealt with the establishment of new cloud computing markets and new opportunities for collaboration in these markets [5, 21, 42]. Collaborative services created by employing cloud services of different vendors enforce the selection and evaluation of these services. We do not want to dive into technical details but propose to use established and proven methods which serve this purpose well. The situation we described resembles issues that have classically been discussed in business administration and operations research: supplier selection. In this field, the situation would be labeled a "new task" buying situation [13]. Hence, we examine the state-of-the-art of methods of supplier selection and evaluate their "fit" for the selection of cloud services in an environment of collaborative cloud markets.

2.1 Literature Review: Methods of Supplier Selection

Of course, there are literature-based comparisons dealing with methods of supplier selection in their respective field (e.g. [1, 14, 23, 26, 49]). However, these reviews do not adhere to the standards of literature reviews as proposed by [46, 50] in the field of (Business) Information Systems. Therefore, we conducted a structured literature review following [46] in order to gain an overview of the methods of supplier selection. We focus on proposed research methods in the context of supplier selection and organize our review in a conceptual way (where the studied concepts are the used methods). We want to highlight central issues and characteristics of these methods for the forthcoming evaluation and transfer to the context of cloud computing.

2.1.1 Conducting the Literature Review

We chose to analyze all journals dealing mainly with the topics of operations research or supply chain management, which are ranked "A" or "B" in the German VHB Jourqual ranking. This ranking is a commonly used, well-documented ranking of journals and conference proceedings in the field of Business Administration [39] and

is easily available online [22]. The list of analyzed journals is displayed in in the appendix[1]. The journals were searched via EbscoHost (database Business Source Premier) and Thomson Reuters's Web of Knowledge. Figure 1 indicates the used search terms and keywords. These keywords were considered to describe the topic comprehensively and have been worked out in an initial explorative search. The search was conducted in September 2012.

TI ("vendor evaluation" OR "vendor selection" OR "vendor choice" OR "Supplier choice" OR "Supplier evaluation" OR "Supplier selection") OR AB ("vendor evaluation" OR "vendor selection" OR "vendor choice" OR "Supplier choice" OR "Supplier evaluation" OR "Supplier selection")

Fig. 1. Search term used for querying EbscoHost

Table 1 in the appendix also shows the number of identified relevant contributions for each journal. In total, we found 225 matching papers which were further analyzed. First, we reviewed title and abstract and second, we scanned the full texts of the papers. We consider all those contributions relevant which make use of, describe or compare specific methods for supplier/vendor selection and evaluation. Finally, we were left with 81 relevant contributions.

2.1.2 Results of the Literature Review

The identified methods in the sample are displayed in Table 2 in the appendix. They are mostly consistent with existing literature reviews on methods of supplier selection (cf. e.g. [23]. In the following, we give a short description, categorization and brief summary of applications of the top three methods. These are the methods used in more than ten per cent of the sample studies: (1) Fuzzy set theory (20.9 %); (2) Analytical Hierarchy Process (19.1 %); (3) Data Envelopment Analysis (10.4 %). A detailed overview (concept matrix) of the methods in the complete sample is given in the appendix. We keep the description of DEA short because later on we do not make use of it.

Fuzzy set theory is based on the seminal paper by Zadeh (1965) [54], which proposes the existence of fuzzy sets. These are classes of "objects with a continuum of grades of membership. Such a set is characterized by a membership (characteristic) function which assigns to each object a grade of membership ranging between zero and one." ([54], p. 338). Fuzzy sets resemble human reasoning and can be used for "dealing with the imprecision intrinsic to many decision problems", as fuzzy set theory is capable of representing vague data ([7], p. 3831). [3] make use of fuzzy sets for supplier selection because the input data for the selection criteria is not known precisely in practice. [11] agree that "in the decision-making process, crisp data may not always be adequate to present the real situation, since human perception, judgment, intuition, and preference remain vague and difficult to measure" ([11], p. 235, similar to [12]) and thus include fuzzy logic in their methodology for supplier

[1] The appendix can be obtained from the authors and is available at: http://www.is.tu-darmstadt.de/media/bwl5_is/forschung/Harnisch-Buxmann-Evaluating_Cloud_Services_Appendix.pdf

choice. [10] incorporate factors of uncertainty, [45] utilize linguistic assessments for supplier evaluation which are transformed into a mathematical decision model by fuzzy logic. [24] state that the fuzzy functions allow the authors to employ human priorities in the decision making process. In the supply chain optimization model by [44] fuzzy (vague) goals are posed and modeled.

Fuzzy methods are often used in combination with AHP (e.g. [6, 7, 9, 29, 30, 53]) and linear or other mathematical programming methods [2, 16, 27, 31, 51]. A detailed list of other combinations can be extracted from the appendix.

The Analytical Hierarchy Process was developed by Thomas L. Saaty [34, 35] and comprises four steps [36]: (1) Definition of the problem; (2) hierarchical structuring of the problem; (3) construction of pairwise comparison matrices. Elements in the upper level are used to compare the elements in the lower level with respect to them. (4) The obtained priorities are used to weigh the priorities in the level immediately below, until the final priorities of the alternatives on the bottom level are obtained.

The elements on the same hierarchy level are usually compared using a 9-point-scale as described by [35] with "1" meaning equal importance of two elements and "9" marking absolute dominance of the first element over the second. Reciprocal judgments are obtained by inversion of the value [33]. [37] shows that eigenvectors are the optimal way to obtain weights or priorities from the pairwise comparison matrices. However, in most cases a simplified method is applied (e.g. [29]) which calculates weights on the basis of averages of column and row sums. [6] state that one advantage of the AHP method is that it is based on pairwise comparisons. Those comparisons can be conducted in a simple way because they are based on an absolute scale [7]. The model structure is easily understandable, flexible [47] and "deeply related to human judgment" ([52], p. 496). It is a widely-used method in the context of many decision problems, especially supplier selection across industries (e.g. chemical processing [32] or telecommunication [43]) and with different goals (e.g. "green" supplier selection [6] or lean procurement [53]). Besides the already mentioned combined methods with fuzzy theory, AHP is applied together with many different methods (e.g. DEA [41, 48, 55]).

Data Envelopment Analysis (DEA) is "an approach for evaluating the efficiencies of decision making units" ([38], p. 744). It was proposed in the seminal paper by Charnes et al. (1978) [8]. Efficiency is usually defined as a ratio of given input values and output values [17]. The resulting model is solved using linear programming [28]. The advantages of DEA are its robustness [23] and "straightforwardness in practical implementation: the DEA approach does not require the decision maker to predefine the criteria weights but these are endogenously determined" ([15], p. 2954.). On the other hand, this is a disadvantage of the method as decision makers do not have control over the importance of criteria [20]. The method itself is varied and personalized in many ways to fit the needs of the respective authors [15].

2.2 Evaluation of Methods for Supplier Selection

In the following, we evaluate the methods for supplier selection which were identified in the literature review in the context of cloud computing. Hence, we define and establish evaluation criteria first, before we evaluate the methods with a simple and easily understandable scheme.

2.2.1 Establishment of Evaluation Criteria

We make use of the following evaluation criteria which are based on the descriptions of the "new task"-buying situation (cf. [13]).

These criteria constitute requirements for a method dealing with the selection of cloud services.

- Need of historical data: The selection method should be capable of supporting decisions without needing historical data of former decisions.
- Effort for evaluation: The effort for the evaluation of services should be manageable; the resulting method must be easily applicable and understandable.
- Qualitative and quantitative data: The method must take qualitative and quantitative data into account.
- Structured method: The method should be structured which relates to effort and understandability. Complex decision problems with many selection criteria should be divided into several, manageable part-decisions.
- Ability to account for uncertainty: Some selection criteria may not be evaluated on a definite basis; there may be "fuzzy" data which the method must cope with.
- Weighting of criteria: Different decision makers have different priorities for the selection criteria; individual weighting of those should be possible.

2.2.2 Evaluation and Resulting Methods

The evaluation was done according to the defined criteria with a simple 0-1-(present/not present=■ or capable/not capable=▬) method. The result is displayed in Table 1.

Table 1. Evaluation of selection methods

Evaluation of methods	No need of historical data	Effort for evaluation	Qualitative and quant. data	Structured method	Ability to account for uncertainty	Weighting of criteria
Fuzzy set theory	■	■	■	▬	■	▬
AHP	■	▬	■	■	▬	■
DEA	■	■	■	■	▬	■/▬

We only show the top three methods from our literature review but also took a look at the other methods. However, we found the three presented methods to be suited best for the purpose.

3 Designing a Novel Approach for the Selection of Collaborative Cloud Services

Our approach is mostly based on AHP. Although there are some disadvantages, it is an often-used and well-proven method for the evaluation and selection of software

(cf. e.g. [25]). We take into account that a decent amount of effort is required to perform an evaluation using AHP and try to mitigate the disability to cope with uncertainty by including some ideas from fuzzy theory within the method.

3.1 High-Level Conceptualization

We make use of AHP's ability to structure complex problems and to decompose them into manageable parts. The hierarchical approach we propose requires the definition of goals on the top-level, describing the most salient aspects of the service which is to be selected. Within these top-level goals (or: categories), we summarize the different criteria of the concrete cloud services that are to be evaluated. An exemplary AHP structure is displayed in Figure 2.

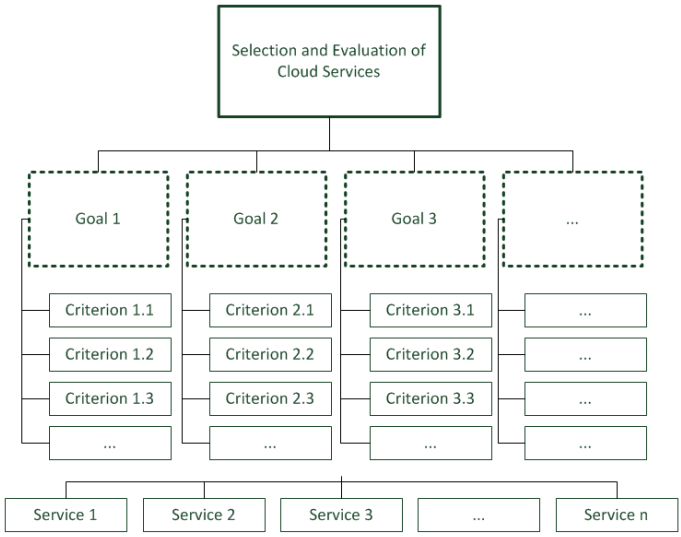


Fig. 2. Exemplary AHP hierarchy

The goals and criteria do certainly differ between different selection problems and can be individually filled for each application of the method. We give an exemplary demonstration in section 3.2. Our approach utilizes fuzzy methods in the pairwise comparison of top-level goals. The decision makers have to evaluate the importance of each category compared to all others on the 9-point-scale for each criterion. Reciprocal judgments (if A is superior to B than B is inferior to A) are obtained by inversion of scores. These scores are "fuzzified" for further calculation. The weighting scheme for the criteria is obtained analogously but uses discrete values. On the lowest level, the alternatives are rated using a simple scoring technique following [30], assigning a value between one and nine to each alternative with respect to each criterion. Using the weights which were calculated before, the ratings of the alternatives for each criterion are integrated so that a final score per alternative can be derived. Of course, it would be possible to employ fuzzy functions in the pairwise

comparisons for the criteria, too. However, we chose simplicity over complexity and believe that the incorporation of aspects of uncertainty regarding categories is sufficient.

The mathematical formulation is given below:

$$\begin{aligned} &n=\text{number of categories, } i = 1,\ldots,n \\ &r_i=\text{number of criteria in category i, } k = 1,\ldots, r_i \\ &m=\text{number of alternatives, } l = 1,\ldots,m \\ &\text{Weights of categories: } W_0 = (q_1, \ldots, q_n) \\ &\text{Weights of criteria in category i: } W_{C_i} = \left(w_{i1}, w_{i2},\ldots,w_{ir_i}\right) \end{aligned} \tag{1}$$

For the rating of alternatives with respect to a certain criterion, we propose to employ a scoring instead of a pairwise comparison approach following [30]. The score of alternatives concerning criteria k in category i is given as:

$$U_l = \left(u_{1ik}, u_{2ik},\ldots,u_{mik}\right) \text{ and } u_{lik} \in \{1,\ldots,9\} \tag{2}$$

With a_{st}=importance of criteria C_s compared to C_t, $s, t \in \{1\ldots n\}$, $a_{ij} \in \{1,\ldots,9\}$, we are able to calculate a vector of weights W_{Ci}

W_{Ci}	C_1	C_2	..	C_1	C_2	…	Weight W_{Ci}
C_1	1	a_{12}	...	$1/\Sigma a_{i1}$	$a_{12}/\Sigma a_{i2}$	…	$(1/\Sigma a_{i1} + a_{12}/\Sigma a_{i2}+\ldots)/n$
C_2	a_{21}	1	...	$a_{21}/\Sigma a_{i1}$	$1/\Sigma a_{i2}$	…	$(a_{21}/\Sigma a_{i1} + 1/\Sigma a_{i2}+\ldots)/n$
…	...	...	1	…	…	…	…
$\Sigma_{s=1}^{r_i}$	a_{i1}	a_{i2}	…	1	1	1	1

(3)

We employ the following fuzzy member function, similar to the one used in [30] which characterizes the membership of z by a triple (l,m,u) with $u \geq m \geq l$ and $l, m, u \in \{1,\ldots,9\}$

$$g(z) = \begin{cases} (1,1,3) & \text{if } z=1 \\ (1,2,4) & \text{if } z=2 \\ (z-2,z,z+2) & \text{if } 3 \leq z \leq 7 \\ (6,8,9) & \text{if } z=8 \\ (7,7,8) & \text{if } z=9 \end{cases} \tag{4}$$

Fuzzy numbers are denoted as $\tilde{a}$ and operations defined as follows:

$$\begin{aligned} \tilde{a}^{-1} &= (l, m, u)^{-1} = \left(\tfrac{1}{u}, \tfrac{1}{m}, \tfrac{1}{l}\right) \\ \tilde{a} \otimes \tilde{b} &= (l_a l_b, m_a m_b, u_a u_b) \\ \tilde{a} \oplus \tilde{b} &= (l_a + l_b, m_a + m_b, u_a + u_b) \end{aligned} \tag{5}$$

We decided to keep our approach simple and to employ the fuzzy function only for the valuation and weighting of categories (goals). This way, uncertainty can be

represented. The pair-wise comparison matrix for the categories is given as a fuzzy matrix of fuzzy pairwise comparison judgments:

$$\left[\tilde{c}_{xy}\right] \text{with } x,y = 1,\ldots,n \tag{6}$$

For the deffuzification several steps are necessary (cf. also [7, 40]). First, the fuzzy synthetic extent $\tilde{S}_x$ with respect to the category x is calculated

$$\tilde{S}_x = \sum_{y=1}^{r_i} \tilde{c}_{xy} \otimes \left[\sum_{k=1}^{r_i}\sum_{y=1}^{r_i} \tilde{c}_{ky}\right]^{-1} \tag{7}$$

Second, the fuzzy ranking values have to be derived (again: [7, 40])

$$V(\tilde{S}_x \geq \tilde{S}_y) = \begin{cases} 1 & \text{if } m_x \geq m_y \\ 0 & \text{if } l_y \geq u_x \\ \dfrac{l_y - u_x}{(m_x - u_x) - (m_y - l_y)} & \text{otherwise} \end{cases} \tag{8}$$

$$V(\tilde{S}_x \geq \tilde{S}_y \mid y=1,\ldots,r_i; y \neq x) = \min_{y \in \{1,\ldots,n\}; y \neq x} V\left(\tilde{S}_x \geq \tilde{S}_y\right), \; x = 1,2,\ldots,n \tag{9}$$

This allows us to finally obtain the vector of weights as given:

$$W_0 = \frac{V(\tilde{S}_x \geq \tilde{S}_y \mid y = 1,\ldots,r_i; y \neq x)}{\sum_{k=1}^{r_i} V(k_x \geq \tilde{S}_y \mid y = 1,\ldots,r_i; y \neq k)}, = \left(q_x\right), \; x = 1,\ldots,n \tag{10}$$

With the weights of categories q_i, the weights of criteria with respect to category i w_{ik} (4) and the scores with respect to alternative l, criteria k and category i u_{lik} (2), we can compute the final scores:

$$S_l = \sum_{i=1}^{n}\sum_{k=1}^{r_i} u_{lik} * w_{ik} * q_i \tag{11}$$

3.2 Illustration of the Method

We want to demonstrate the proposed method using a simple example, given a case with two suitable services A and B which are to be evaluated against three goals.

1. Network access: Criteria availability and encryption
2. Measured service: Criteria: price and license model
3. Self-service: usability and adaptability

The following calculation steps are necessary:

1) Establish the matrix of pairwise comparisons for the top-level categories

$$\begin{pmatrix} 1 & \frac{1}{3} & 7 \\ 4 & 1 & \frac{1}{5} \\ \frac{1}{7} & 5 & 1 \end{pmatrix} \rightarrow \text{Fuzzy matrix} \begin{pmatrix} (1,1,3) & (\frac{1}{8},\frac{1}{6},\frac{1}{4}) & (5,7,9) \\ (4,6,8) & (1,1,3) & (\frac{1}{7},\frac{1}{5},\frac{1}{3}) \\ (\frac{1}{9},\frac{1}{7},\frac{1}{5}) & (3,5,7) & (1,1,3,) \end{pmatrix}$$

2) Defuzzification using (7-10) $\rightarrow \sum_{y=1}^{3} \tilde{c}_{xy} = \begin{pmatrix} (1+4+\frac{1}{9}, 1+6+\frac{1}{7}, 3+8+\frac{1}{5}) \\ (\frac{1}{8}+1+3, \frac{1}{6}+1+5, \frac{1}{4}+3+7) \\ (5+\frac{1}{7}+1,\ 7+\frac{1}{5}+1,\ 9+\frac{1}{3}+3) \end{pmatrix}^{T}$

and $\left[\sum_{k=1}^{3}\sum_{y=1}^{3} \tilde{c}_{ky}\right]^{-1} = (0.030, 0.046, 0.065)$

$\rightarrow \tilde{S}_x = \begin{pmatrix} (0.15, 0.33, 0.73) \\ (0.27, 0.4, 0.67) \\ (0.40, 0.53, 0.8) \end{pmatrix}^{T}$

$V(S_{Network} \geq S_{Measured}) = 1.0$
$V(S_{Network} \geq S_{Self}) = 0.69$
$V(S_{Measured} \geq S_{Network}) = 1$
$V(S_{Measured} \geq S_{Self}) = 0.87$
$V(S_{Self} \geq S_{Network}) = 1$
$V(S_{Self} \geq S_{Measured}) = 1$

And thus, the final weights for the categories are $W_0 = (0.27, 0.34, 0.39)$

3) The pairwise comparison matrices for the criteria are given as follows:

Network $\begin{pmatrix} 1 & 5 \\ \frac{1}{5} & 1 \end{pmatrix}$ Measured $\begin{pmatrix} 1 & \frac{1}{3} \\ 3 & 1 \end{pmatrix}$ Self $\begin{pmatrix} 1 & 7 \\ \frac{1}{7} & 1 \end{pmatrix}$

4) The scores for the alternative services with respect to the criteria are:

	Service A	Service B	Weight of criteria w_{ik} (using formula (3))
Availability	5	3	0.83
Encryption	7	9	0.17
Price	3	5	0.25
License model	8	3	0.75
Usability	5	5	0.875
Adaptability	8	9	0.125

This allows for the calculation of the final scores using (11):
Alternative A: 5.83; alternative B: 4.41. Hence, service A should be selected.

4 Conclusion, Limitation and Future Work

We developed and illustrated a method for the evaluation and selection of cloud services. As the method is rather generic, it is applicable in the context of collaborative cloud services, as well as for the selection of standalone services. The method allows decision makers to incorporate fuzzy logic up to a certain amount and can thus account for uncertainty. As we did not want to complicate things further than necessary, we restricted the usage of fuzzy logic to the goals within the AHP process. An extension to the criteria-level is certainly possible.

Our research has some limitations: The literature review we conducted is subjective to a certain extent: The filtering process was carried out manually. We propose one design of a method for the evaluation of services, but there is a manifold of possible fuzzy AHP designs to choose from. We strived to present an understandable but still comprehensive idea. Further research could cope with the modification and extension of the proposed method, in order to automatize some of the manual evaluation and decision phases. Another idea might be to design a decision support system which helps decision makers in the evaluation of several cloud services. We demonstrated the applicability of methods from the field of supplier selection to the business information systems field and hope to encourage further research in this area. We can certainly learn and profit from a vast body of knowledge which another discipline has accumulated over decades.

Acknowledgement. The work presented in this paper was performed in the context of the Software-Cluster project InDiNet (www.software-cluster.org). It was funded by the German Federal Ministry of Education and Research (BMBF) under grant no. "01IC10S04". The authors assume responsibility for the content.

References

1. Aissaoui, N., Haouari, M., Hassini, E.: Supplier selection and order lot sizing modeling: A review. Computers & Operations Research 34, 3516–3540 (2007)
2. Amid, A., Ghodsypour, S.H., O'Brien, C.: Fuzzy multiobjective linear model for supplier selection in a supply chain. International Journal of Production Economics 104, 394–407 (2006)
3. Amid, A., Ghodsypour, S.H., O'Brien, C.: A weighted max–min model for fuzzy multi-objective supplier selection in a supply chain. International Journal of Production Economics 131, 139–145 (2011)
4. Armbrust, M., Fox, A., Griffith, R., Joseph, A.D., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I.: A view of cloud computing. Communications of the ACM 53, 50–58 (2010)
5. Biao, S., Hassan, M.M., Eui-Nam, H.: A Novel Heuristic-Based Task Selection and Allocation Framework in Dynamic Collaborative Cloud Service Platform. In: 2010 IEEE Second International Conference on Cloud Computing Technology and Science (CloudCom), pp. 360–367 (2010)
6. Büyüközkan, G.: An integrated fuzzy multi-criteria group decision-making approach for green supplier evaluation. International Journal of Production Research 50, 2892–2909 (2012)

7. Chan, F.T.S., Kumar, N., Tiwari, M.K., Lau, H.C.W., Choy, K.L.: Global supplier selection: a fuzzy-AHP approach. International Journal of Production Research 46, 3825–3857 (2008)
8. Charnes, A., Cooper, W.W., Rhodes, E.: Measuring the efficiency of decision making units. European Journal of Operational Research 2, 429–444 (1978)
9. Che, Z.H.: A genetic algorithm-based model for solving multi-period supplier selection problem with assembly sequence. International Journal of Production Research 48, 4355–4377 (2010)
10. Chen, C.-M.: A fuzzy-based decision-support model for rebuy procurement. International Journal of Production Economics 122, 714–724 (2009)
11. Chen, L.Y., Wang, T.-C.: Optimizing partners' choice in IS/IT outsourcing projects: The strategic decision of fuzzy VIKOR. International Journal of Production Economics 120, 233–242 (2009)
12. Chou, S.-Y., Shen, C.-Y., Chang, Y.-H.: Vendor selection in a modified re-buy situation using a strategy-aligned fuzzy approach. International Journal of Production Research 45, 3113–3133 (2007)
13. De Boer, L., Labro, E., Morlacchi, P.: A review of methods supporting supplier selection. European Journal of Purchasing & Supply Management 7, 75–89 (2001)
14. Degraeve, Z., Labro, E., Roodhooft, F.: An evaluation of vendor selection models from a total cost of ownership perspective. European Journal of Operational Research 125, 34–58 (2000)
15. Dotoli, M., Falagario, M.: A hierarchical model for optimal supplier selection in multiple sourcing contexts. International Journal of Production Research 50, 2953–2967 (2012)
16. Faez, F., Ghodsypour, S.H., O'Brien, C.: Vendor selection and order allocation using an integrated fuzzy case-based reasoning and mathematical programming model. International Journal of Production Economics 121, 395–408 (2009)
17. Falagario, M., Sciancalepore, F., Costantino, N., Pietroforte, R.: Using a DEA-cross efficiency approach in public procurement tenders. European Journal of Operational Research 218, 523–529 (2012)
18. Ferrer, A.J., Hernández, F., Tordsson, J., Elmroth, E., Ali-Eldin, A., Zsigri, C., Sirvent, R., Guitart, J., Badia, R.M., Djemame, K.: OPTIMIS: A holistic approach to cloud service provisioning. Future Generation Computer Systems 28, 66–77 (2012)
19. Frischbier, S., Gesmann, M., Mayer, D., Roth, A., Webel, C.: Emergence as Competitive Advantage. In: Maciaszek, L., Cuzzocrea, A., Cordeiro, J. (eds.) Proceedings of the 14th International Conference on Enterprise Information Systems, ICEIS 2012, vol. 3, pp. 181–186. INSTICC, Wroclaw (2012)
20. Golmohammadi, D., Creese, R.C., Valian, H., Kolassa, J.: Supplier Selection Based on a Neural Network Model Using Genetic Algorithm. IEEE Transactions on Neural Networks 20, 1504–1519 (2009)
21. Hassan, M., Song, B., Huh, E.-N.: A market-oriented dynamic collaborative cloud services platform. Annals of Telecommunications 65, 669–688 (2010)
22. VHB JOURQUAL 2.1, http://vhbonline.org/en/service/jourqual/vhb-jourqual-21-2011/
23. Ho, W., Xu, X., Dey, P.K.: Multi-criteria decision making approaches for supplier evaluation and selection: A literature review. European Journal of Operational Research 202, 16–24 (2010)
24. Humphreys, P., McCloskey, A., McIvor, R., Maguire, L., Glackin, C.: Employing dynamic fuzzy membership functions to assess environmental performance in the supplier selection process. International Journal of Production Research 44, 2379–2419 (2006)

25. Jadhav, A.S., Sonar, R.M.: Evaluating and selecting software packages: A review. Information and Software Technology 51, 555–563 (2009)
26. Kaufmann, L., Carter, C.R., Buhrmann, C.: Debiasing the supplier selection decision: a taxonomy and conceptualization. International Journal of Physical Distribution & Logistics Management 40, 792–821 (2010)
27. Kumar, M., Vrat, P., Shankar, R.: A fuzzy programming approach for vendor selection problem in a supply chain. International Journal of Production Economics 101, 273–285 (2006)
28. Kuo, R.J., Lin, Y.J.: Supplier selection using analytic network process and data envelopment analysis. International Journal of Production Research 50, 2852–2863 (2012)
29. Labib, A.W.: A supplier selection model: a comparison of fuzzy logic and the analytic hierarchy process. International Journal of Production Research 49, 6287–6299 (2011)
30. Lee, A.H.I.: A fuzzy AHP evaluation model for buyer-supplier relationships with the consideration of benefits, opportunities, costs and risks. International Journal of Production Research 47, 4255–4280 (2009)
31. Lin, R.-H.: An integrated model for supplier selection under a fuzzy situation. International Journal of Production Economics 138, 55–61 (2012)
32. Pitchipoo, P., Venkumar, P., Rajakarunakaran, S.: A distinct decision model for the evaluation and selection of a supplier for a chemical processing industry. International Journal of Production Research 50, 4635–4648 (2012)
33. Saaty, R.W.: The analytic hierarchy process—what it is and how it is used. Mathematical Modelling 9, 161–176 (1987)
34. Saaty, T.L.: Axiomatic foundation of the analytic hierarchy process. Management Science 32, 841–855 (1986)
35. Saaty, T.L.: How to make a decision: the analytic hierarchy process. European Journal of Operational Research 48, 9–26 (1990)
36. Saaty, T.L.: Decision making with the analytic hierarchy process. International Journal of Services Sciences 1, 83–98 (2008)
37. Saaty, T.L., Vargas, L.G.: Inconsistency and rank preservation. Journal of Mathematical Psychology 28, 205–214 (1984)
38. Saen, R.F.: Suppliers selection in the presence of both cardinal and ordinal data. European Journal of Operational Research 183, 741–747 (2007)
39. Schrader, U., Hennig-Thurau, T.: VHB-JOURQUAL2: method, results, and implications of the German Academic Association for business research's journal ranking. BuR - Business Research 2, 180–204 (2009)
40. Şen, C.G., Şen, S., Başlıgil, H.: Pre-selection of suppliers through an integrated fuzzy analytic hierarchy process and max-min methodology. International Journal of Production Research 48, 1603–1625 (2010)
41. Sevkli, M., Koh, S.C.L., Zaim, S., Demirbag, M., Tatoglu, E.: An application of data envelopment analytic hierarchy process for supplier selection: a case study of BEKO in Turkey. International Journal of Production Research 45, 1973–2003 (2007)
42. Siebenhaar, M., Lampe, U., Lehrig, T., Zöller, S., Schulte, S., Steinmetz, R.: Complex service provisioning in collaborative cloud markets. In: Abramowicz, W., Llorente, I.M., Surridge, M., Zisman, A., Vayssière, J. (eds.) ServiceWave 2011. LNCS, vol. 6994, pp. 88–99. Springer, Heidelberg (2011)
43. Tam, M.C.Y., Tummala, V.M.R.: An application of the AHP in vendor selection of a telecommunications system. Omega 29, 171 (2001)
44. Tsai, W.-H., Hung, S.-J.: A fuzzy goal programming approach for green supply chain optimisation under activity-based costing and performance evaluation with a value-chain structure. International Journal of Production Research 47, 4991–5017 (2009)

45. Vahdani, B., Zandieh, M.: Selecting suppliers using a new fuzzy multiple criteria decision model: the fuzzy balancing and ranking method. International Journal of Production Research 48, 5307–5326 (2010)
46. Vom Brocke, J., Simons, A., Niehaves, B., Riemer, K., Plattfaut, R., Cleven, A.: Reconstructing the giant: on the importance of rigour in documenting the literature search process. In: 17th European Conference on Information Systems, pp. 1–13 (2009)
47. Wang, G., Huang, S.H., Dismukes, J.P.: Product-driven supply chain selection using integrated multi-criteria decision-making methodology. International Journal of Production Economics 91, 1–15 (2004)
48. Wang, Y.-M., Chin, K.-S., Leung, J.P.-F.: A note on the application of the data envelopment analytic hierarchy process for supplier selection. International Journal of Production Research 47, 3121–3138 (2009)
49. Weber, C.A., Current, J.R., Benton, W.C.: Vendor selection criteria and methods. European Journal of Operational Research 50, 2–18 (1991)
50. Webster, J., Watson, R.T.: Analyzing the Past to Prepare for the Future: Writing a Literature Review. MIS Quarterly 26, 1–11 (2002)
51. Wu, D.D., Zhang, Y., Wu, D., Olson, D.L.: Fuzzy multi-objective programming for supplier selection and risk modeling: A possibility approach. European Journal of Operational Research 200, 774–787 (2010)
52. Xia, W., Wu, Z.: Supplier selection with multiple criteria in volume discount environments. Omega 35, 494–504 (2007)
53. Yu, M.-C., Goh, M., Lin, H.-C.: Fuzzy multi-objective vendor selection under lean procurement. European Journal of Operational Research 219, 305–311 (2012)
54. Zadeh, L.A.: Fuzzy sets. Information and Control 8, 338–353 (1965)
55. Zhang, X., Lee, C.K.M., Chen, S.: Supplier evaluation and selection: a hybrid model based on DEAHP and ABC. International Journal of Production Research 50, 1877–1889 (2012)

Towards User Interface Patterns for ERP Applications on Smartphones

Marcus Homann, Holger Wittges, and Helmut Krcmar

Technische Universität München, Chair for Information Systems, Boltzmannstr. 3, 85748 Garching, Germany
{marcus.homann,holger.wittges,krcmar}@in.tum.de

Abstract. Aim of this paper is to identify recurring user interface tasks of Enterprise Resource Planning applications for smartphones. The identified tasks are used to build reusable user interface design patterns. The proposed patterns aim to accelerate the application design process and improve the user experience across different ERP applications through a consistent user interface. Twenty existing ERP applications for smartphones were analyzed using a task analysis approach. The resulting task trees are examined in terms of recurring, cross-application user interface tasks. The results revealed that the examined applications are mainly focused on selecting, presenting and manipulating business objects. The identified tasks were used to build corresponding user interface patterns. Finally, the patterns were structured through a pattern catalogue.

Keywords: Mobile, Smartphone, ERP, Enterprise Resource Planning, User Interface Pattern.

1 Introduction

The technological progress in the area of mobile devices and the improvement of mobile data networks facilitated new application possibilities. In particular, the proliferation of smartphones is rapidly increasing [1]. Smartphones differ from other advanced mobile phones through their variety of integrated sensors, like accelerometer-, gyroscope-, compass-, optical proximity-, ambient-light sensor, camera- or global positioning system (GPS)-sensor. Most smartphones rely on touchscreens for the primary interaction with users. In addition, specialized operating systems like iOS or Android are installed on smartphones. These operating systems expose application programming interfaces (APIs) that allow the implementation of applications that offer new functionality to users [2, 3].

While the application development for smartphones was initially focused on the consumer market, there is a gaining interest in the business market. According to a CIO survey of Gartner [4] 61% of the respondents plan to enhance their efforts regarding mobile technologies during the next three years. Enterprise Resource Planning (ERP) systems are one of the major software systems in companies [5]. Mobile access to those systems offers the potential to improve business processes [6, 7]. Usability and the

W. Abramowicz (Ed.): BIS 2013, LNBIP 157, pp. 14–25, 2013.

corresponding user interfaces (UIs) are recognized success factors of software systems [8]. However, the UI design of smartphone applications is challenging due to the smaller screen sizes and limited data input possibilities [9]. A multitude of smartphone applications that access ERP data and –functionality is possible due to the huge amount of functionality in ERP systems [10]. However, there is a risk that similar tasks are implemented differently for diverse applications. But, consistency is an important aspect to achieve a good usability [8, 11]. Users are accustomed to perform tasks in a certain way. However, the UI of traditional ERP applications on desktop computers is criticized [12, 13].

UI patterns are one way to collect reusable UI design knowledge. They focus on a certain design problem and provide a structured description of the solution in a pre-defined format [14]. Using UI patterns can increase the consistency across different applications. Existing UI patterns for smartphone applications mainly focus on general UI design problems like presenting content on smaller screens, navigating between different screens of the application or supporting different screen sizes [9, 15, 16]. They often use examples of consumer applications like online stores or social networks. Specific UI problems of enterprise applications like smartphone ERP applications are hardly considered. UI patterns represent proven design solutions [9]. Therefore, this study examines available ERP applications for smartphones of the leading ERP vendor SAP in order to identify reusable UI patterns.

The paper is structured as follows: the next section characterizes smartphone ERP applications. Then, related research papers are presented. In the following, existing smartphone ERP applications are analyzed using a task analysis in order to identify recurring tasks. Then, UI patterns are introduced as reusable solutions for recurring UI design problems. Moreover, a pattern catalogue with 16 UI patterns for the development of smartphone ERP applications is illustrated. Finally, the paper concludes with a brief summary and outlook.

2 Characteristics of Smartphone ERP Applications

In this paper, smartphone ERP applications are understood as applications that are installed on smartphones and access functionality and data from ERP systems through offered interfaces. Aim of this section is to demonstrate the specifics of this application domain. Therefore, the general characteristics of smartphone applications are named first, followed by characteristics of ERP systems.

2.1 Characteristics of Smartphone Applications

Smartphone applications are generally focused on a certain use case and are written with less source code than desktop applications [17]. Unlike desktop applications, smartphone applications generally need to be more responsive to sensor data [17]. Examples are device movements or data from the GPS sensor. In addition, smartphone screens offer less space to display content. Thus, it is not possible to display multiple windows at the same time, like on desktop applications. Moreover, special controls that can be used with fingers on touchscreens must be provided as

well as the reaction to smartphone typical gestures [9]. In addition, the vendors of operation systems for mobile devices provide guidelines for designing mobile applications, especially for the UI [17]. Furthermore, data entries through a smartphone on-screen keyboard are less efficient than with classical keyboards [18].

2.2 Characteristics of ERP Systems

ERP systems aim to enable a consistent, uniform data storage and support integrated business processes across multiple functional areas of companies [19, 20]. Typical application areas are manufacturing, sales, accounting, finance and human resources. The most widely installed ERP systems are currently SAP Business Suite[1] und Oracle Applications[2].

ERP systems are usually built on a client-server architecture and comprise four layers [19]. This includes the data management-, application-, adaption- and user interface layer. The data management layer contains the database server as well as access to external data sources like web services. The application layer embraces the implemented application programs that are usually referred to as transactions. The application functionality is generally offered through APIs to third-party applications. Examples are the business application programming interfaces (BAPIs) of the SAP Business Suite. The adaption layer allows the customization of the underlying layers in order to support the specifics of the business process and data structures of companies. The UI layer comprises the client programs for accessing the ERP system by users. ERP system vendors often provide native- as well as web-based user interfaces.

The applications of ERP systems are largely focused on processing business objects [19]. Typical business objects of ERP systems are for example customer-, material-, or sales order objects. Usual processing activities on business objects are create-, read- update- and delete operations. The workflow of ERP systems often involves the manipulation of several business objects in a specific order. For example a sales process involves the creation of an inquiry, followed by a quotation and finally a sales order. In addition, there is a dependency between business objects. For example, the mentioned inquiry-, quotation and sales order objects are related to a customer object.

ERP systems have an extensive functionality and are therefore often referred to as complex application systems [5]. In addition, the usability of their UIs is criticized [12, 13, 21]. Typical problem areas are the high search effort to identify the required functionality, inadequate support during transaction execution and in error situations and the inconsistent usage of UI elements across multiple transactions.

2.3 ERP Applications on Smartphones

Based on the described characteristics of smartphone applications, existing ERP applications cannot be converted to smartphone applications [10]. Desktop ERP UIs

[1] http://www.sap.com/germany/solutions/business-suite/index.epx
[2] http://www.oracle.com/us/products/applications/overview/index.html

often comprise several tabs, each with a variety of data entry fields. Figure 1 illustrates a typical UI of an ERP desktop application; in this case to create sales orders.

The conversion of such UIs to smartphone applications would lead to unhandy smartphone applications. Therefore, design strategies should be identified to build smartphone applications with usable UIs that meet their focused tasks.

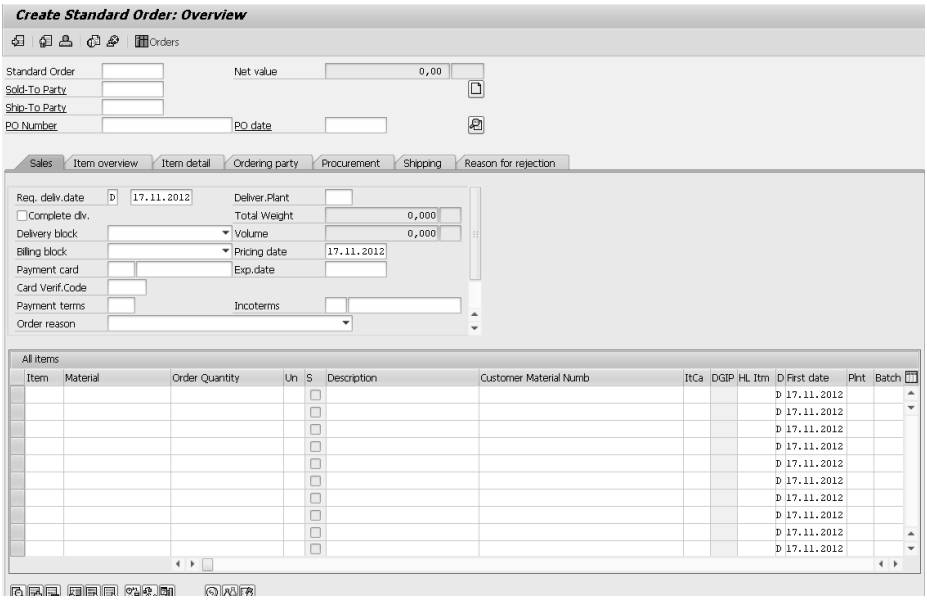


Fig. 1. Example of an UI of an ERP desktop application to create sales orders

3 Related Research

There is little research available that deals with the UI design specifics of smartphone ERP applications. Kurbel et al. [22] investigate how ERP content can be displayed on mobile device screens. They propose three adaption mechanisms: content adaption, adaption of style as well as layout and structural adaption. The authors also suggest a couple of UI design principles for mobile ERP applications like minimizing user input.

Kurbel and Dabkowski [6] explore how ERP content can be accessed and displayed on heterogeneous mobile devices. The authors propose a two layered approach, consisting of a component that transforms content into an XML format and a component that transforms the XML content into various user interface representations, depending on the mobile device characteristics.

Brans and Basole [23] propose categories of reusable software components for developing mobile enterprise applications. They revealed that mobile enterprise applications have in common that they work on information objects, which mostly offer the operations read, create, update and alert.

4 Task Analysis of Existing ERP Applications for Smartphones

In the following, twenty ERP applications for smartphones were analyzed using the task analysis approach according to Paternò [24]. Aim is to identify recurring UI tasks within these applications.

Task analysis is an established method to examine interactive applications [24]. The purpose of a task analysis is to identify the tasks and their properties that an application should support to reach user goals. Task models illustrate the user's interaction with an application. The results are captured in so called task models. ConcurTaskTrees (CTT) is a notation to specify task models [24]. CTT uses a graphical syntax, a hierarchical structure and supports temporal relationships between tasks [24]. Tasks Analysis and CTT are used in this paper to analyze and describe existing ERP applications for smartphones.

Twenty ERP applications for smartphones were selected for the analysis. Only applications of the ERP vendor SAP were examined. In addition, only those applications were considered that were available on 4th September 2012 in Apple's App Store. By selecting applications from SAP, which were accepted in Apple's App Store, a certain quality regarding the usability is assumed. On 4th September 2012 forty-three applications were available from SAP. Firstly, only those applications were selected that have an ERP focus. In addition, only applications that offered a demo mode were considered. This reduced the effort to test the application, because no additional software components and configuration steps were necessary. After this selection, the amount of applications to be analyzed comprised twenty applications.

Figure 2 illustrates one of the created CTT models using the employee lookup application as example. The application offers tasks to display the application users' employee profile as well as other employee profiles. For the later task the search requires the surname and/or the family name of the employee of interest. After entering this information by the user, the application displays the search results and the user is able to select one employee entry. In the following, the attributes of the selected employee are shown, including the position, office address and contact information. It is possible to contact the displayed employee by SMS, E-Mail or to establish a phone call. It is also possible to add the selected employee profile to a favorites list.

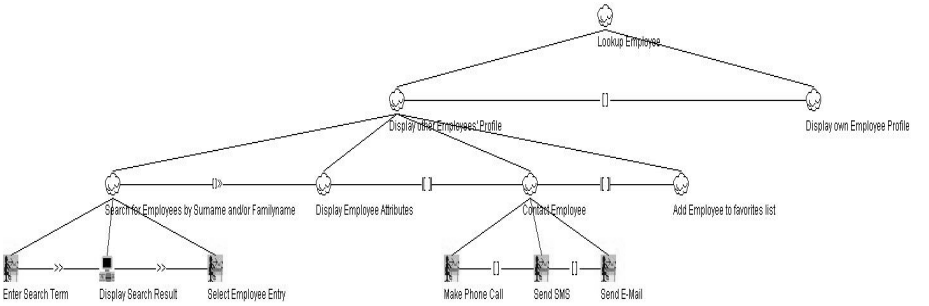


Fig. 2. ConcurTaskTree of the Employee Lookup Application

The overall results of the task analysis are shown in table 1. The first column includes the name of the applications, as mentioned in Apple's App Store, and the examined version number of the application in parentheses. The second column lists all business objects that could be identified across all analyzed applications. The third column includes the sum of all recurring tasks identified across all analyzed applications.

Table 1. Identified business objects recurring tasks

Name of the ERP application	**Business Object**	**Recurring Task**
- Business One (1.6.0) - Business ByDesign (3.5.1) - Sales order notification (2.2.0) - Customers and contacts (2.2.0) - Sales order notification (2.2.0) - Retail execution (2.1.1) - Timesheet (2.3.0) - Employee lookup (2.2.0) - Material availability (2.2.0) - Transport tendering (1.2.0) - Travel expense approval (2.2.1) - Travel expense report (1.0.1) - Travel receipt capture (2.2.1) - Payment approvals (2.2.0) - Quality issue (1.1.0) - In-Store product lookup (1.0.1) - ERP order status (2.2.0) - Customer financial fact sheet (3.0.2) - HR approvals (2.3.0) - Leave request (2.3.1)	- Time entry - Travel expense - Business trip - Business partner - Customer - Contact - Activity - Sales opportunity - Sales quotation - Sales offer - Sales order - Product - Price - Invoice - Material - Quality issue - Service call - Service contract	- Select business object type - Get list of all business objects - Sort business object list by attribute value - Group business object list by attribute value - Search for business objects by attribute value - Filter business object list by attribute value - Display business object Attributes - Navigate to related business object - Create business object - Change business object attributes - Delete business object - Add business object to favorites list

The result reveals that even though the analyzed smartphone applications focus on different business objects, the offered tasks are similar. More comprehensive applications like Business One or Business ByDesign start with the selection of a business object type like sales orders or customers (select business object type). Subsequently, applications typically list all business objects of the focused business object type (get list of all objects). In some cases the listing is grouped by a certain attribute (group business object list by attribute value). For example travel requests are grouped according their status in open, approved or rejected. A couple of applications offer the functionality to show only business objects with a certain attribute value by defining a filter (filter business object list by attribute value).

An example is to list only sales orders of a particular customer. In many cases, there is also the possibility to search for certain business objects through a keyword (search for business objects by attribute value). In most cases a certain business object can be selected from the search result to display its attribute values (display business object attributes). Partially, the values of these attributes can be changed within the smartphone application (change business object attributes). In some applications it is also possible to create new business objects (create business object) or delete existing ones (delete business object).

Overall, the possible tasks for different business object types are very similar. In addition to the task itself, it is revealed that the tasks follow a logical order. For example the task "select business object type" is mostly followed by a "get list of all business objects" task; update and delete object tasks are usually only possible after a "display business object attributes" task. Because of this similarity from a task perspective, it reasonable to implement these tasks with similar UI representations in order to achieve consistency across different ERP applications.

4.1 Creating User Interface Patterns for Mobile ERP Applications

The analysis of the UI representations of the selected applications revealed that the recurring tasks are often implemented with similar combinations of UI elements. Generally, list- and form-based UI representations are often used. List-based UI representations are generally used to display the output of search- and filter tasks. Form-based UI representations are a collection of textfields with corresponding labels. They are often used to display business object attributes or to create new business objects. However, there are also inconsistencies between the selected applications. An example is the task "filter business object list by attribute value". It is partially implemented by a selection list, partially by a tab bar. Another example is the content and layout of the single cells of a selection list. There are applications that display only a single identification number; others show additional data in different fonts and partially with the use of special symbols. These different UI implementations of similar tasks can lead to user confusion. The provision of UI patterns for smartphone ERP applications for designers and programmers of smartphone ERP applications can reduce these potential inconsistencies.

4.2 User Interface Patterns

Collected design knowledge about user UIs is usually provided in form of guidelines, principles or patterns [14]. Principles such as the eight golden rules of Shneiderman [11] have a generic character, while guidelines are often platform specific [25]. Both tend to be difficult to use during the design process, because they suggest absolute validity [14]. On the contrary, UI patterns explicitly focus on context and are thus problem related [14]. They tell the designer when, how and why the provided solution is applicable [14].

The concept of design patterns was originated in urban architecture by Christopher Alexander [26], adapted to object-oriented software design by Beck and Cunningham [27] and became popular through the book of Gamma et al. [28]. Generally spoken, patterns are structured descriptions of an invariant solution to a recurrent problem in context [29]. They follow a three-part rule, which expresses the relation between a certain context, a problem and a solution [26]. The interest in patterns in Human-Computer Interaction dates back to 1994 [30]. According to a CHI workshop in 2000 an UI pattern "captures the essence of a successful solution to a recurring usability problem in interactive systems" [25]. Therefore patterns can be seen as descriptions of best practices which capture common solutions to design tensions and are thus by definition not novel [9].

4.3 A User Interface Pattern Catalogue for Mobile ERP Applications

In our research, we have developed 16 UI patterns for smartphone ERP applications so far. In order to improve the overview, we have structured our patterns in a pattern catalogue. The catalogue structure follows a typical navigation structure in smartphone ERP applications that is selection – presentation – interaction. Most applications require the user to select the business object of interest first (selection). Afterwards the attributes of the selected business object are shown (presentation). In some cases it is also possible to manipulate the selected business object (interaction). Figure 3 illustrates our current smartphone ERP application UI pattern catalogue.

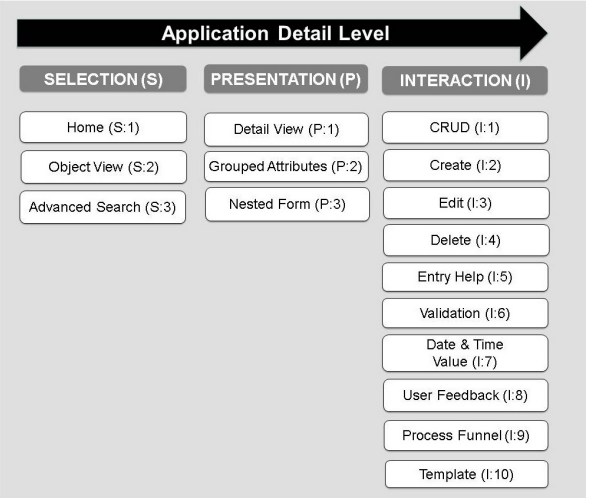


Fig. 3. User Interface Pattern Catalogue for Smartphone ERP applications

The single UI patterns are constructed according to the UI pattern structure proposed by Tidwell [9]: pattern name, what, use when, why, how, examples.

- **What:** this section provides a short description of the pattern.
- **Use when:** this section explains the context of use where the pattern can be applied. In our case, this is generally a description of the focused recurring task.
- **Why:** this section describes why the pattern should be used.
- **How:** describes the solution and how it solves the focused problem
- **Examples:** illustrates examples how the pattern has been successfully applied, usually in form of screenshots.

In the following, only one selected pattern of our catalogue is presented due to space limitations. On interest, the descriptions of the remaining patterns can be requested from the authors via email.

4.4 Example Pattern: Nested Forms

Name:
Nested Form Pattern

What:
A nested navigation between a summarized listing of a business object and its attributes or a navigation between related business objects.

Use when:
This pattern should be primarily used to establish a navigation link between a selection screen (listing, search, filter) and the screen to display the attributes of a selected business object. It should be also used to navigate between related business objects.

Why:
Due to limited screen size of smartphones it is generally not possible to display the attributes of more than one business object. In order to solve this problem the Nested Form Pattern should be used.

How:
To implement this pattern a navigation link between the summarized business object and its detail view must be implemented. The summarized business object should be presented in a selectable cell. It should have an indicator in form of an arrowhead to indicate the cell is selectable. There should be a navigation link to the detail view when selecting the cell. There should be also a button on the top of the details screen that gives the user the possibility to navigate backwards.

Examples:
This example illustrates the nested forms pattern using the SAP materials availability application for the iPhone as example. The left screens illustrated a selection screen with different materials. By selecting one material its attributes are shown. It is possible to leave the attributes screen by selecting the back button.

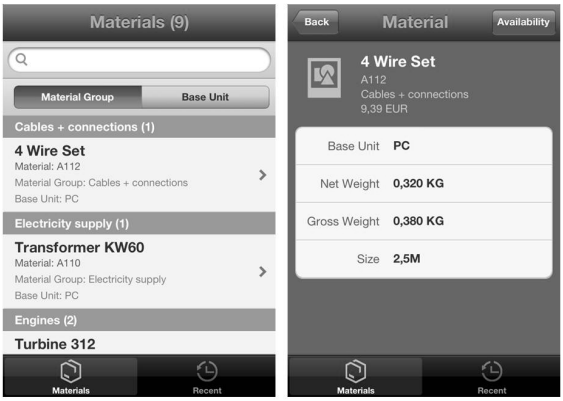


Fig. 4. Nested Forms Patterns illustrated within the SAP Materials Availability iPhone Application

5 Summary and Outlook

In this paper, twenty selected ERP applications for smartphones have been analyzed using a task analysis. The analysis revealed twelve recurring tasks that are offered across different applications. Based on these tasks, a pattern catalog of currently sixteen patterns for the UI design of smartphone ERP applications was developed. The usage of these patterns for future smartphone ERP applications is able to increase the consistency and thus the usability of the corresponding applications.

However, the current UI patterns are based on existing applications. The authors suspect that there will be improvements of existing applications in the near future, which will affect the presented patterns. One indicator for the dynamics in this area is the high rate of change of the applications during the analysis. One area of improvement could be the application functionality. The current ERP applications for smartphones offer significantly less functionality than the corresponding desktop applications. For example, search forms of desktop applications allow complex searches across a combination of different attributes, while the smartphone applications usually provide only one search box. The overall challenge is to extend the functionality of current smartphone applications but ensure a good usability.

Current smartphone ERP applications hardly use sensor data. A few exceptions are photographing barcodes to find material or photographing vouchers of travel expenses. Thus, there is unused potential to reduce the effort of data input. The current pattern catalogue is focused on iPhone UIs. Because the Android UI is pretty similar, we expect that the adaption to Android is not much effort. However, other mobile operation systems like Windows Phone have a different interaction style and also different UI elements. Thus, the current UI patters may not be applicable for Windows Phone. These areas should be considered in future extensions and improvements of the presented patterns.

References

1. Gartner: Hype Cycle for Mobile Device Technologies 2012. G00234209, Stamford, CO, USA (2012)
2. Krannich, D.: Mobile System Design - Herausforderungen, Anforderungen und Lösungsansätze für Design, Implementierung und Usability-Testing Mobiler Systeme. Books on Demand, Norderstedt (2010)
3. Tri Do, T.M., Blom, J., Gatica-Perez, D.: Smartphone Usage in the Wild: a Large-Scale Analysis of Applications and Context. In: 13th International Conference on Multimodal Interfaces, pp. 179–194 (2010)
4. Gartner: Gartner Executive Programs Amplifying the Enterprise: The 2012 CIO Agenda, http://www.gartner.com/it/page.jsp?id=1897514 (accessed at: July 16, 2012)
5. Kurbel, K.: Enterprise Resource Planning und Supply Chain Management in der Industrie. Oldenbourg, Munich (2011)
6. Kurbel, K., Dabkowski, A.: A multi-tier architecture for mobile enterprise resource planning. In: Wirtschaftsinformatik, Dresden, Germany, pp. 75–93 (2003)
7. Mladenova, V., Homann, M., Kienegger, H., Wittges, H., Krcmar, H.: Towards an Approach to Identify and Assess the Mobile Eligibility of Business Processes. In: American Conference of Information Systems, Detroit, USA, pp. 1–11 (2011)
8. Nielsen, J.: Usability engineering. Academic Press, Boston (1993)
9. Tidwell, J.: Designing interfaces, 2nd edn. O'Reilly, Sebastopol (2011)
10. Mall, S., Stefanov, T., Stadelman, S.: Mobilizing your Enterprise with SAP. SAP Press, Boston (2012)
11. Shneiderman, B., Plaisant, C.: Designing the user interface: strategies for effective human-computer interaction, 5th edn. Addison-Wesley, Boston (2010)
12. Topi, H., Lucas, W., Babaian, T.: Identifying Usability Issues with an ERP Implementation. In: International Conference on Enterprise Information Systems, pp. 128–133 (2005)
13. Oja, M.-K., Lucas, W.: Evaluating the Usability of ERP Systems: What can critical incidents tell us? In: Fifth Pre-ICIS Workshop on ES Research, pp. 1–6 (2010)
14. van Welie, M., van der Veer, G.C., Eliens, A.: Pattern Languages in Interaction Design. In: International Workshop on Tools for Working with Guidelines (2000)
15. Nilsson, E.G.: Design patterns for user interface for mobile applications. Advances in Engineering Software 40, 1318–1328 (2009)
16. Neil, T.: Mobile Design Pattern Gallery. O'Reilly, Sebastopol (2012)
17. Wasserman, A.I.: Software engineering issues for mobile application development. In: Proceedings of the FSE/SDP Workshop on Future of Software Engineering Research, pp. 397–400 (2010)
18. Ballard, B.: Designing the mobile user experience. John Wiley, West Sussex (2007)
19. Gronau, N.: Enterprise Resource Planning: Architektur, Funktionen und Management von ERP-Systemen. Oldenbourg, Munich, (2010)
20. Krcmar, H.: Informationsmanagement, 5th edn. Springer, Heidelberg (2010)
21. Singh, A., Wesson, J.: Evaluation criteria for assessing the usability of ERP systems. In: Proceedings of the 2009 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists on - SAICSIT 2009, pp. 87–95 (2009)
22. Kurbel, K., Jankowska, A.M., Nowakowski, K.: A mobile user interface for an ERP system. Issues in Information Systems 7, 146–151 (2006)

23. Brans, P.D., Basole, R.C.: A comparative anatomy of mobile enterprise applications: Towards a framework of software reuse. Information Knowledge Systems Management 7, 145–158 (2008)
24. Paternò, F.: Model-based design and evaluation of interactive applications. Springer, London (2000)
25. Borchers, J.: A pattern approach to interaction design. Wiley, New York (2001)
26. Alexander, C., Ishikawa, S., Silverstein, M.: A pattern language: towns, buildings, construction. Oxford University Press, New York (1977)
27. Beck, K., Cunningham, W.: Using Pattern Languages for Object-Oriented Programs. In: Workshop on Specification and Design for Object-Oriented Programming (1987)
28. Gamma, E., Helm, R., Johnson, R., Vlissides, J.: Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley (1995)
29. Dearden, A., Finlay, J.: Pattern Languages in HCI: A Critical Review. Human-Computer Interaction 21, 49–102 (2006)
30. Rijken, D.: The Timeless Way... the design of meaning. SIGCHI Bulletin 6, 10 (1994)

TrustAider – Enhancing Trust in e-Leadership

Yue Dai[1], Calkin Suero Montero[1], Tuomo Kakkonen[1], Mohsen Nasiri[1], Erkki Sutinen[1], Mina Kim[2], and Taina Savolainen[2]

[1] School of Computing, University of Eastern Finland, Länsikatu 15
80110 Joensuu, Finland
{yvedai,calkins,tuomo.kakkonen,erkki.sutinen}@uef.fi,
mohsen@student.uef.fi
[2] Department of Business, University of Eastern Finland
{mina.kim,taina.savolainen}@uef.fi

Abstract. Trust in leadership is significantly influenced by the current IT dominated business environment. We introduce TrustAider, a model for supporting trust building in a business environment through the use of text analysis methods and an interactive user interface. TrustAider integrates natural language processing technologies to provide feedback and suggestions on how to interact in a way that enhances mutual trust during the process of computer mediated communication between leaders, employees, and customers. This paper presents an overview of the TrustAider's architecture and its key components. The effective functionalities of TrustAider for promoting trust are also demonstrated with a use case.

Keywords: trust in e-leadership, natural language processing, sentiment analysis, content analysis.

1 Introduction

Trust is defined as beliefs and expectancies by an individual or a group so that they can rely on the word, promise, or any verbal or written statement of another individual or group [1]. Trust is an essential resource and skill for effective leadership in business environment, and it has gained increasing attention in organizational practice and research [2]. *Trust building* helps to develop mutual respect, openness, understanding, and empathy; it is a demanding and bilateral process that requires mutual commitment and effort.

Strong culture of trust in leadership influences the climate in an organization towards vitality and innovativeness, and helps in lowering the risks inherent in the organization and transaction costs of commercial actions. For example, trust in Steve Jobs was seen as an important factor that influenced the success of Apple Inc. [3]. Trust in leadership can, moreover, provide opportunities to access knowledge, and to improve performance such as creativity, problem solving and proactive implementation of new ideas [4, 5].

The increasing global dispersion of organizations and the explosion of the use of *information and communication technologies* (ICT) have changed the way leaders and

W. Abramowicz (Ed.): BIS 2013, LNBIP 157, pp. 26–37, 2013.

followers interact with each other within organizations and between organizations. The term '*e-leadership*' refers to these changes, and it involves a social influence process supported by ICT where changes are brought about in attitudes, feelings, thoughts, behaviors, styles, way of working and performance of individuals, groups, and organizations [2, 6]. For example, *computer-mediated communications* (CMCs) in the form of email and information management systems have changed the way in which businesses store and disseminate their vital knowledge. ICT can support the communication, and the collection and dissemination of information required to sustain, improve and transform organizational work. Communication problems and misunderstanding in CMCs are, however, seen as the most common issue for trust building in e-leadership [2, 7, 8, 9]. At the same time, trust building in e-leadership is different from the traditional way that is predominantly based on face-to-face interactions.

In order to support trust building in e-leadership, we propose the TrustAider model, which is based on automatic *content analysis*, *sentiment analysis* (SA), and *information extraction* (IE) tools among other *natural language processing* (NLP) technologies. Through TrustAider, computer mediated communications between leaders, employees and customers are expected to become more efficient and effective. By directly improving the quality of written communication, TrustAider will, in turn, contribute to trust building [2, 6, 7, 8, 9].

2 Background

2.1 Computer Mediated Communications and Trust Building

Literature shows that CMCs are a means to support trust within online business activities. Pavlo and Dimoka [10] showed how the online feedback system of marketplaces (e.g., eBay) impacts the buyers' trust in the seller's benevolence and credibility, which are integral components of trust. Pavlo and Dimoka used manual content analysis of over 11,000 text comments to classify them into 5 categories: outstanding/abysmal benevolence, outstanding/abysmal credibility, and ordinary. The results of the study showed that "text comments had greater impact on a seller's credibility and benevolence than did crude numerical ratings" [10]. Hence, this form of computer mediated communication (i.e., online feedback) possesses a significant economic value as more buyers will implicitly trust online sellers that have greater outstanding feedback comments. These comments, as a result, contribute to the seller's trustworthiness.

To improve user's trust in online systems, Pu and Chen [11] proposed the creation of organization-based explanation interfaces, i.e., interfaces in which the best matching item of a product search is displayed at the top of the interface, with several alternatives alongside it. This kind of interfaces showed to be "highly effective to build users' trust" in an online recommendation system [11]. The organization-based explanation interface showed to foster competence-inspired trust in the buyer based on its easiness of use and its efficiency in comparing products.

Both of these studies used different forms of CMCs to approach different aspects of trust building between customers and sellers within online commercial environments.

Our work differs from the above studies in that we focus on aiding trust building not only between customers and organizations but also among employees themselves, and between employees and their leaders. The analysis of computer mediated communications with automatic NLP methods and technologies is also a differentiating point of our work. As explained below, we draw from the factors of *clearness of the contents* in CMCs for trust building in organizations given by O'Bryan (1995) and *trustworthiness* defined by Mayer et al. (1995) [2, 12, 13, 15].

2.2 Clearness of Communication Content

Due to the significant increase in information, knowledge and network cooperation, globalization and CMC have changed the requirements for efficient trust building [7, 8, 9]. In several studies, *communication quality* has been identified to influence trust building [2, 6, 8, 12]. Various communication quality attributes have been suggested; among them clearness, accuracy, currency, and credibility play an important role. The clearness of communication content, that was introduced by O'Brien (1995), implies that if the content of the communication is clear and precise to describe the issue at hand, then such content can foster a trusting relationship [13, 14]. We focus our attention on improving the clearness of communication as a means for building trust, as this characteristic can in itself contribute to the credibility of the parties involved.

2.3 Trustworthiness

Mayer et al. [15] considered positive expectation for other's trustworthiness trait as a significant antecedent for trust and suggested three factors of trustworthiness: ability, benevolence and integrity. *Ability* is defined as "the group of skills, competencies and characteristics that enable a party to have influence within some specific domain". Research has shown that perceived expertise is an essential and critical characteristic of the trustee (a party to be trusted) for trust [16, 17]. *Benevolence* is "a perception of a positive orientation of the trustee to the trustor (a trusting party), including intentions, expressions of genuine concern and care, aside from an egocentric profit motive". Characteristics of leaders, employees, and customers such as altruism, loyalty, and considering the subordinates' needs and desires are all related to benevolence [18, 19]. *Integrity* is the "perception that the trustee adheres consistently to a set of principles acceptable to the trustor, such as honesty and fairness". For example, the consistency actions of leaders/employees, their credible communications and congruent actions with their words, along with the belief that the trustee has a strong sense of justice are influential features to perceive integrity [15]. TrustAider aims at improving these factors as detailed below.

3 TrustAider

The TrustAider model supports trust building in organizations by improving the clearness and trustworthiness of communications with the help of NLP technologies. Figure 1 describes the framework of TrustAider. As Figure 1 illustrates, the input data consists of an organization's CMCs, for example, emails, instant messages, monthly reports,

decision reports, and texts published through social media sites. State-of-the-art NLP technologies will be integrated into TrustAider to detect several features of the input text and give writers feedback on three main areas: a) how to improve text fluidity and clarity as to become easy to understand for readers from different cultural backgrounds, b) how to make the text more positive, and c) how to make the key points of the message more explicit so that it can be understood more effectively and efficiently. Using TrustAider, the content of the CMCs among leaders, among employees, between employees and leaders, and between organizations and customers will be clearer; the readers will understand the main ideas of a given message and the contributions of the sender easily. Hence, by improving the clearness of communication TrustAider can improve trust within organizations and help promote better e-leadership.

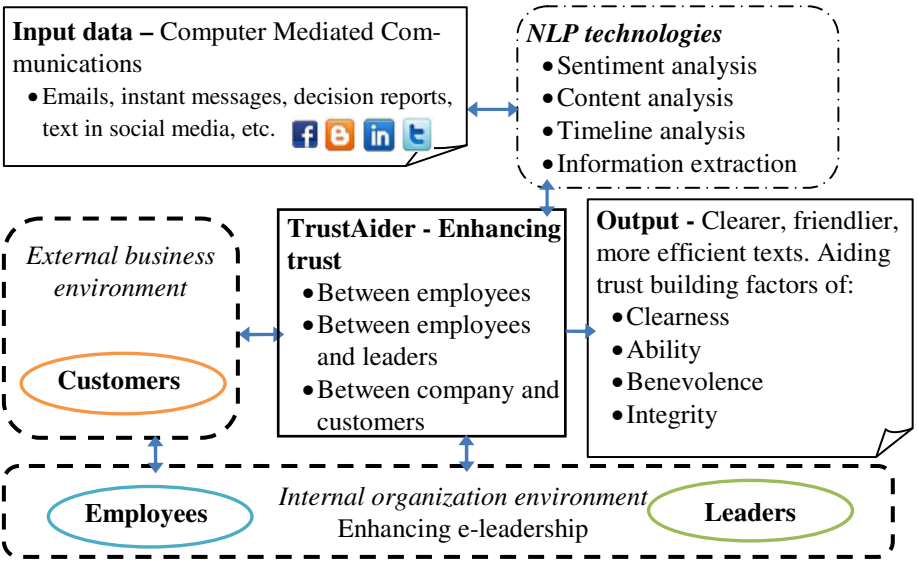


Fig. 1. The framework of TrustAider

3.1 Core Technologies of TrustAider

Table 1 shows the relation between the factors of trustworthiness and clearness of communication content and the core technologies used by TrustAider.

Content Analysis. Text fluidity can affect how easily a text is understood; the more fluid the connections between sentences are, the less memorizing is needed to understand a given written message. Hence, text fluidity improves the clearness of CMCs and, in turn, helps trust building [2, 6, 8, 13, 14]. TrustAider incorporates text fluidity detection in its content analysis process in order to search for possible topic discontinuities within and across paragraphs. Three levels of text fluidity will be used to assess an input text [20]: *a) High* - a sentence is consistent with previous sentences; *b) Low* - a sentence is connected to a previous one, but the connection is through one or

more unconnected sentences, or the connection is only apparent at the very end of the paragraph; *c) None* - a sentence is not connected to any of the previous sentences, or there are paragraphs in between.

TrustAider also uses automatic content analysis in the form of IE to mine key concepts or features (keywords) in the input text that are related to the abilities and competence of the trustees. Proficiency in technical areas of expertise and knowledge of scientific terms are example of such features. These key features are considered as important parts of the messages and are reused in the output message in a later stage.

The style of the text is also analyzed via content analysis technology. Generally a piece of written text can be either formal or informal according to the type of expressions it contains, e.g., passive voice is used more often in formal language style whereas active voice is preferred by an informal style [21, 22]. Within a professional environment, the usage of a more formal language style to communicate with leaders could positively reflect on the competencies of a person.

Table 1. Relation between trust building factors and TrusAider technologies

Factor		Technology	Functions	Desired Output	Description
Clearness of communication content		Content analysis	Text fluidity detection	Fluidity level	*High, low, none*
Trustworthiness	Ability		Extracting key concepts	Summarized	The length of text is *shorter*. Main concepts are explicit. This saves the readers' time.
			Expression	Formal	Emphasis on *formal* expressions.
				Informal	Emphasis on *informal* expressions.
	Benevolence	Sentiment analysis	Sentiment level	Friendly	Sentiment content is *high*. More sentiment words are suggested to users.
				Factual	Sentiment content is *low*. None or less use of sentiment words.
	Integrity	Timeline analysis	Response time tracking	Statistics and graphs	Over a long period, average response time can be interpreted as an indication of the level of integrity in relation to work.
		Information extraction	Excuse detection	List and graphs based timelines	How often are excuses detected?

Sentiment Analysis. Benevolence shows how trustees feel towards the organization they work for and towards their jobs. SA aims to process ongoing communications inside the organization, and determine the sentiment orientation or polarity of the

messages. This can promote reaching mutual awareness between readers and writers [23], and can foster awareness of the current sentiment polarity of the text (positive/neutral/negative) and how this polarity could be changed in order to promote trust. In this manner leaders can be aware of the benevolence that the employees express in text as well as the benevolence express in the leaders' communications. Through TrustAider the user can change the sentiment level of the communication to a friendly or to a factual level depending on the needs of the situation.

Timeline Analysis. TrustAider will detect signs of the user's integrity implicitly contained in CMC messages. Integrity refers to trustees' principles and accepted moral ethics, enabling organizations and leaders to believe in their employees' honesty and the morality of their work [6, 15]. Here we consider *the when and how a person responds to emails* as signs of the level of their integrity, since this reflects the ethics of the workplace. By tracking the time when communications are answered, TrustAider can estimate work integrity, as this indicates the effort a person put in their work and how important their job is for them.

Information Extraction. The fact that an employee or a leader tends to make excuses for late completion of assigned tasks also reflects on the person's ethics and can also signal a lack of work integrity [6, 15]. By extracting keywords and phrases that contain excuses (e.g., "sorry, I forgot", "it won't happen again", and so forth), TrustAider can detect excuses in text. Information extraction technology is also used during the process of content analysis to detect important keywords related to the competencies of a person, as explained previously. With a clear image of their people's integrity and competencies toward the work they are assigned to do, leaders can make more robust decisions and trust can improve.

3.2 TrustAider Architecture

Figure 2 illustrates the architecture of TrustAider that we are currently implementing. An explanation to the TrustAider's architectural components follows.

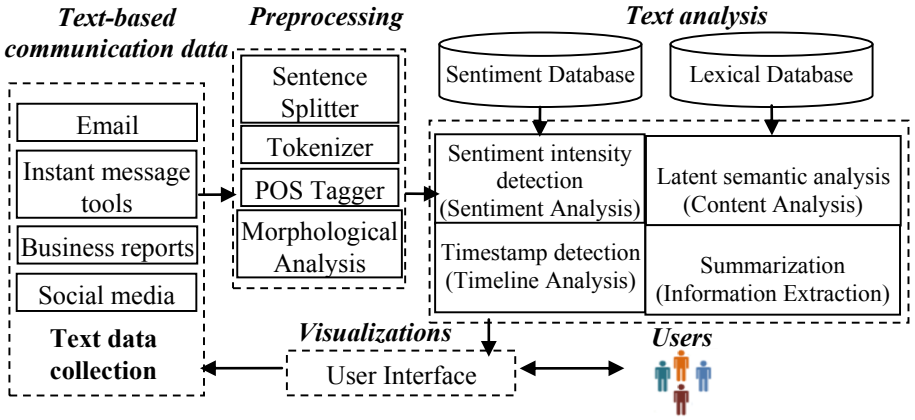


Fig. 2. TrustAider architecture

Text-Based Communication Data. This component holds the TrustAider's input. The system is designed to support text-based CMCs, hence, the input data can be any piece of written document written by the leader or the employee of a business organization.

Preprocessing. This component holds a cascade of NLP algorithms used in order to understand the input text. The Sentence Splitter process segments documents into sentences to aid sentence-level analysis. The Tokenizer then separates the sentences into their basic constituents or tokens, such as words, numbers, symbols and punctuation. A part-of-speech (POS) tagger then marks each token with its correspondent identification based on the token's context. The POS tagger identifies tokens as, for example, nouns, verbs, adjectives, adverbs and so on. Finally, a morphological analyzer extracts the root and affixes of the identified tokens. The aim of the preprocessing is to determine the role and the meaning of each token in respect to other tokens within a sentence. These morphologically tagged tokens will then be used during the content analysis process within the text analysis component.

Text Analysis. This component holds the TrustAider technologies outlined in Table 1. During the text analysis process the following information is extracted: proper names (e.g., Human Resources, iPad, John Doe); person's title (e.g., manager, CEO, Mr., Dr., MBA, etc.); tasks (e.g., launching a new product or performing a market survey); phrases that contain excuses (e.g., "sorry, I forgot", "it won't happen again"); sentiment words (such as "good", "bad", used to infer the positive/neutral/negative polarity of the text); and sentiment intensity. The information extracted is then used by TrustAider core technologies as explained here.

The *timeline analysis*, records explicit temporal information, such as the date when a computer-mediated communication (CMC) was sent or received, alongside the names of the sender and receiver. It also records implicit information through a timestamp: date of an event such as launching a new product, or time taken to complete a task, for instance submitting a survey report. *Temporal expressions* (TE), "the day before that", "last quarter", or "yesterday", are used to describe the temporal location of a task. Using timestamps, tasks, events and detected excuses are placed on a timeline, allowing temporal references to be automatically resolved. *Latent semantic analysis* (LSA) is used to carry out the content analysis of the input text. LSA looks for similarities between a set of documents (or sentences) as to analyze relationships among the members of the set. Since LSA gives a score for similarity, it is relevant to determine the fluidity and formality of the analyzed text. Based on this TrustAider is able to present appropriate synonyms candidates to specific words and phrases in order to increase the text fluidity and formality as required by the user. The *sentiment intensity detection* process implements a SA algorithm to identify the polarity (positive/neutral/negative) of words and phrases. The intensity of the sentiment is also ranked alongside its polarity, such as positive (+1 to +3), negative (-3 to -1), and neutral (0). This ranking is based on a sentiment database composed of sentiment-bearing words and phrases. The process of *summarization* pulls out key information from the preprocessed text, such as tasks, events, proper names and titles. This key information is then reconstructed into syntactically correct sentences through the use of a lexical database containing word synonyms and semantic relations between different words. The information is then presented to the user in a more concise format.

The factors of trustworthiness and clearness of communication content are effectively addressed by TrustAider as follows:

(a) Clearness of communication factor: improving text fluidity and formality using LSA.
(b) Ability factor: summarizing text and presenting its information in a more concise and formal manner.
(c) Benevolence factor: promoting benevolence-awareness as outlined by the sentiment intensity detection process.
(d) Integrity factor: as given by the timeline analysis of the input CMCs.

Visualization. The TrustAider interface graphically presents the results of the input text analysis, showing the text fluidity and sentiment levels, along with the text expression type (i.e., formal/informal). Once a text is thus analyzed, TrustAider can provide guidance on how to improve it, as to convey the desired level of trust. For example a user may wish to change the sentiment level of the text or increase its fluidity. When interacting with TrustAider, users can choose a function on the interface (e.g., text fluidity, sentiment level, excuse detection) and a desired output type according to their needs. Additionally, the text processed through a function in the interface can be processed again through a different function recursively. For example after analyzing the fluidity of a text, the user may want to analyze the text again this time looking at the sentiment level of the message, and so forth.

4 Use Case for TrustAider

Let us reflect on an organization in which thousands of email messages are being exchanged daily between the leaders and the employees, and among the employees themselves. Employees send work reports to their supervisors; supervisors inform managers of their progress; and major results are sent to the company's CEO. It is well documented that the tone of the communication at each level of the organization's hierarchy is different, and that every day a significant amount of working time is spent on how to make an email or report more easy-to-understand for the leaders or the co-worker [24]. TrustAider can improve this situation in several ways. For example, when an employee wants to send a piece of CMCs to her supervisor, she writes the message and then processes it through TrustAider. Figure 3 shows the processes flow of TrustAider for a given written text.

Let us consider a hypothetical message: *"Hello Mr. Martinez. Work on the report file is done. This month we did more sales. We had problems in the marketing section because of the art department. We didn't have creative designers. We have to get more money for our section to hire more people. Complete report file is attached."*

Here, to simulate the TrustAider system interface, functionality and behavior we used already built individual tools. These tools are not yet integrated into one single system, but their output analysis have been cascaded to demonstrate the TrustAider workings. After the text analysis, TrustAider will be able to conclude that the above message has an overall negative sentiment and that it contains elements of informal language (Figure 4).

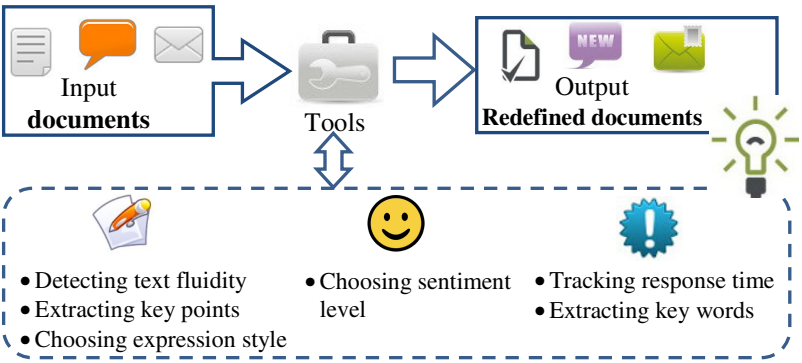


Fig. 3. The process flow of TrustAider

From: John Doe
To: Manager Smith
Subject: report file
Sent: Friday, March 26, 2010 9:11AM

Hello Mr. Martinez. Work on the report file is done. This month we did more sales. We had problems in the marketing section *because of* the art department. We *didn't have* creative designers. We have to *get* more money for our section to hire more people. Complete report file is attached.

Fluidity level:	**Sentiment:**			**Expression:**
High Low None	Positive +1	Positive +2	Positive +3	Formal *Informal*
	Negative -1	Negative -2	Negative -3	**Timestamp**: March 2010
	Neutral 0			

Fig. 4. Example of a TrustAider input text analysis. The fluidity levels were generated by the SWAN system [20]; the sentiment analysis result was given by SentiStrength (`http://sentistrength.wlv.ac.uk/`); and the formality was evaluated based on a word list [25]

TrustAider could improve this example by increasing its formality and fluidity whilst preserving the intended meaning. The message given in Figure 5 is a candidate output. As this figure illustrates, the meaning of message remains the same, however, through content analysis, the style of the language has been changed to a more formal and factual one. The words *due (to)*, *lack (of)* and *budget* were manually chosen from a formal language glossary [25] and are replacing the informal phrases *because (of)*, *didn't have* and *get (more money)*. The sentiment of the message, although not directly modified, has been positively improved by omitting straightforward negations (i.e., *didn't*) and replacing the introductory greeting *Hello* with *Dear*. This creates a message that is formal, charged with a more positive tone. Content analysis also helps TrustAider to establish semantic relations between different words and sentences which can be joined together. For instance, TrustAider may conclude that the two sentences containing the keywords *report files* are referring to the same subject, and

that connecting them through the grammatical conjunction and could lead to a concisely summarized text. This leads to a message that has clear and explicit key points, making its content more easily understood and its meaning efficiently delivered. The timeline analysis, in turn, clarifies temporal expressions within the input text. With the reference frame set to the date of sending the email, for this use case the timestamp will be given by the sentence: *"*Work on the report file is done. *This month we did more sales" (Timestamp: March, 2010).*The fluidity level analysis, sentiment intensity detection, expression style and timeline analysis are automatically performed by the corresponding system's components. Currently we are developing the automatic summarization component of TrustAider, the output here is a manually simulated candidate.

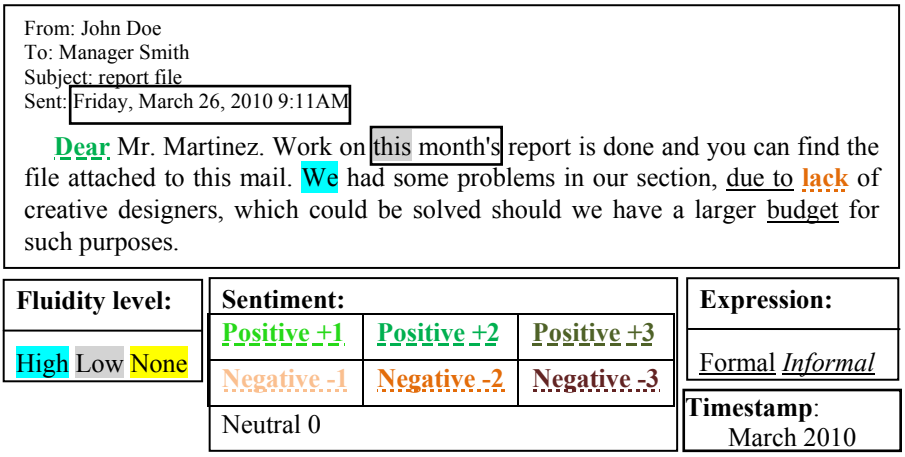


Fig. 5. Example of a TrustAider improved output. The fluidity levels were generated by the SWAN system [20]; the sentiment analysis result was given by SentiStrength (`http://sentistrength.wlv.ac.uk/`); and the formality was evaluated based on a word list [25].

The timeline will show the activity of the person indicated in the average response time. Using this information, the person's integrity can be assessed [6, 15]. It is worth noting that the process of improving a piece of written text as shown here is an interactive one: while the core processes of TrustAider are automatically performed, the user has ultimate control in deciding the shape of the final output. TrustAider can assist improving the text's language style among other features; however the user should lead each stage of the process. We have demonstrated with this use case TrustAider's functionality to improve the text fluidity, to make it more positive, and to clarify its key points.

We have observed that TrustAider can identify the text fluidity and sentiment polarity in an acceptable level comparing to the results of manual work. But for detecting the excuses and expression styles, the performance need be improved. The expectations for TrustAider from the user's perspective, the user interface can be designed more friendly. The results could be reported in other formats than the way demonstrated in this case.

Although the system integration needs further development, in the kind of situations depicted in the above case study the TrustAider architecture successfully identifies the text fluidity and sentiment polarity of the input text. Hence, TrustAider useful feedback can lead to improvement in the clearness of communication content as indicated in the text fluidity; the ability as perceived by the writing style of the user; and the benevolence-awareness as indicated by the sentiment intensity of the text.

5 Conclusion and Future Plan

We have outlined TrustAider, a system for enhancing trust in e-leadership through the use of various natural language processing technologies. We showed that by combining content analysis, sentiment analysis, information extraction and timeline analysis, TrustAider is able to efficiently analyze and suggest improvements to computer mediated communications. These improvements target to enhance the clearness of communication and the three factors of trustworthiness: ability, benevolence, and integrity. We have discussed the TrustAider's general framework and system architecture, and also have explained how NLP technologies can be implemented to improve trust building in e-Leadership. We also demonstrated via a use case how TrustAider could benefit trust building within a business scenario.

Our ongoing work on TrustAider involves finalizing the integration of a fully functional system and evaluating it in more varied business environment situations. Ethical use of TrustAider will require transparency within the organization, that is, all users should be aware of TrustAider's functions and capabilities.

Acknowledgements. The research reported in this article was supported by the following research projects: "Detecting and Visualizing Emotions in Texts" a project, which is funded by the Academy of Finland and "Towards e-leadership: higher profitability through innovative management and leadership systems" a project which is funded by the European Regional Development Fund and TEKES – the Finnish funding agency for technology and innovation.

References

1. Deutsch, M.: Cooperation and Trust: Some Theoretical Notes. In: Jones, M.R. (ed.) Nebraska Symposium on Motivation, pp. 275–320. University of Nebraska Press, Oxford (1962)
2. Savolainen, T.: Trust Building in e-Leadership – Important Skill for Technology-Mediated Management in the 21st Century. In: Proceedings of the International Conference on Management, Leadership and Governance. Academic Conference Limited, England (2013)
3. Welter, F.: All You Need is Trust? A Critical Review of the Trust and Entrepreneurship Literature. International Small Business Journal 30, 193–212 (2012)
4. Ford, C.M., Gioia, D.A.: Factors Influencing Creativity in the Domain of Managerial Decision Making. Journal of Management 26, 705–732 (2000)
5. Parker, S.K., Williams, H.M., Turner, N.: Modeling the Antecedents of Proactive Behavior at Work. Journal of Applied Psychology 91, 636–652 (2006)
6. Mackenzie, M.L.: Manager Communication and Workplace Trust: Understanding Manager and Employee Perceptions in the E-world. International Journal of Information Management 30, 529–541 (2010)

7. Bos, N., Olson, J., Gergle, G., Olson, G., Wright, Z.: Effects of Four Computer-Mediated Communications Channels on Trust Development. In: Proceedings of the SIGCHI Conference on Human Factors in Computer Systems, pp. 135–140. ACM, USA (2002)
8. Ridings, C.M., Gefen, D., Arinze, B.: Some Antecedents and Effects of Trust in Virtual Communities. Journal of Strategic Information Systems 11, 271–295 (2002)
9. Vasalou, A., Hopfensitz, A., Pitt, J.V.: In Praise of Forgiveness: Ways for Repairing Trust Breakdowns in One-off Online Interactions. International Journal of Human-Computer Studies 66, 466–480 (2008)
10. Pavlou, P.A., Dimoka, A.: The Nature and Role of Feedback Text Comments in Online Marketplaces: Implications for Trust Building, Price Premiums, and Seller Differentiation. Information Systems Research 17, 243–268 (2006)
11. Pu, P., Chen, L.: Trust Building with Explanation Interfaces. In: Proceedings of the 11th International Conference on Intelligent User Interfaces, pp. 93–100. ACM, USA (2006)
12. Govindarajan, V., Gupta, A.K.: Building an Effective Global Business: Successful Teams Strive to Build Trust and Overcome Barriers of Geography, Language and Culture. MIT Sloan Management Review 42, 63–71 (2001)
13. O'Brien, R.C.: Is Trust a Calculable Asset in the Firm? Business Strategy Review 6, 39–54 (1995)
14. Sydow, J.: Understanding the Constitution of Interorganizational Trust. In: Lane, C., Bachman, R. (eds.) Trust Within and Between Organizations: Conceptual Issues and Empirical Applications. Oxford University Press, UK (1998)
15. Mayer, R.C., Davis, J.H., Schoorman, F.D.: An Integrative Model of Organizational Trust. Academy of Management Review 20, 709–734 (1995)
16. Sitkin, S.B., Roth, N.L.: Explaining the Limited Effectiveness of Legalistic "Remedies" for Trust/Distrust. Organization Science 4, 367–392 (1993)
17. Hovland, C.I., Janis, I.L., Kelley, H.H.: Communication and Persuasion, New Haven, CT. Yale University Press, USA (1953)
18. Frost, T., Stimpson, D.V., Maughan, M.R.C.: Some Correlates of Trust. Journal of Psychology: Interdisciplinary and Applied 99, 103–108 (1978)
19. Butler, J.K., Cantrell, R.S.: A Behavioral Decision Theory Approach to Modeling Dyadic Trust in Superiors and Subordinates. Psychological Reports 55, 19–28 (1984)
20. Kinnunen, T., Leisma, H., Machunik, M., Kakkonen, T., Lebrun, J.L.: SWAN – Scientific Writing AssistaNt: A Tool for Helping Scholars to Write Reader-Friendly Manuscripts. In: The 13th Conference of the European Chapter of the Association of Computational Linguistics, pp. 20–24. ACL, Avignon (2012)
21. Centre for English and Language Communication, National University of Singapore: Using Appropriate Words in an Academic Essay, http://www.nus.edu.sg/celc/research/books/cwtuc/chapter03.pdf (accessed January 19, 2013)
22. Business Language Services SRL. Formal and Informal English, http://www.blssrl.com/assets/Formal_Informal_English.pdf (accessed January 19, 2013)
23. Liu, B.: Sentiment Analysis and Opinion Mining. Morgan & Claypool Publishers, USA (2012)
24. The Social Economy: Unlocking Value and Productivity through Social Technologies. Online at MacKinsey Global Institute, http://www.mckinsey.com/insights/mgi/research/technology_and_innovation/the_social_economy (accessed January 21, 2013)
25. Word List (Formal and Informal Words for Emails), https://writingclinic.pbworks.com/w/page/33438334/Word%20list%20(formal%20and%20informal%20words%20for%20emails) (accessed January 21, 2013)

The ERP App Store: Diverging and Converging Stakeholder Interests in a PaaS Ecosystem

Andreas Nilsson[1] and Johan Magnusson[2]

[1] Center for Service Science & Innovation, Stockholm University, DSV,
Forum 100, SE-164 40 Kista, Sweden
[2] Centre for Business Solutions, University of Gothenburg
Box 100, S-405 30 Gothenburg, Sweden

Abstract. This study addresses stakeholder perceptions in Platform as a Service (PaaS) ecosystems. Inherent in the platform strategy is dependence between those stakeholders that own the platform and those supplying the added functionality. This is investigated through an in-depth case study of a one ERP Vendor's implementation of a platform strategy. Through employing the technology of app stores, the vendor intends to shift from delivering software as a service (SaaS) to PaaS. The study identifies and discusses relational issues impacted by the new business model emerging from the platform strategy. The results show that the introduction of a platform strategy brings with it the threat of restructuring the business model and power relationships nested in the ecosystem. This is discussed from a platform governance perspective.

Keywords: ERP, Platform strategy, App store, PaaS, Business model, Governance.

1 Introduction

As noted by Martens and Hamerman (2011), the market for packaged software has been under transition since the introduction of cloud-based delivery models. Vendors of monolith solutions, such as SAP and Oracle are experiencing pressure from smaller vendors, offering products with a more narrow functional scope (Magnusson et al, 2012). The mantra of these challenging vendors is criticizing the one-vendor policy, advocating a best-of-breed strategy, fuelled by technological developments within cloud computing for off-site hosting and dynamic sizing (Böhm et al, 2011).

With this technological development comes a decrease in integration cost, and, new business models such as cloud service brokerage (CSB), software as a service (SaaS), integration as a Service (IaaS), and, platform as a Service (PaaS) (Ghormley, 2012). The latter of these models (PaaS) is reported having a particular allure to vendors of ERP and CRM systems (Ku and Cho, 2011). With the commercial success of mobile platforms such as Apple's AppStore, Google's Android Market and Blackberry's AppWorld, vendors from the packaged software industry have vamped their efforts to explore the "app store" potential. Examples such as AppExchange from Salesforce.com and Marketplace from Lawson show an emerging trend where

W. Abramowicz (Ed.): BIS 2013, LNBIP 157, pp. 38–49, 2013.

vendors of packaged software are moving towards becoming PaaS actors, often through the creation and control of a marketplace for platform related apps (Nambisan and Sawhney, 2011; Ghormley, 2012; Kim, Kim and Lee, 2010; Martens and Hamerman, 2011).

Despite the strong stream of research within packaged software (Grabski et al, 2011; Sarker et al, 2012), there has so far been only a limited amount of literature addressing issues related to app stores and platform ecosystems (see Gawer and Cusumano (2002) and Burkardt et al (2012) for notable exceptions). The objective of this study is to add new insight to the phenomena of app stores for packaged software. This objective will be guided by the following research question:

Which differences among stakeholder's perceptions of optimal governance can be found in a PaaS ecosystem?

The central contribution of this study is an empirical identification of differences in perception between two categories of stakeholders in the platform ecosystem (Vendors and Partners) in terms of the optimal settings of governance. The remainder of the paper starts with a review of previous studies related to the areas covered in this paper. After that, we present the method and theoretical framework (§3), followed by the results (§4) and a discussion of the findings (§5).

2 Previous Research

2.1 From ERP systems to Platform as a Service

Packaged software have since the mid 1990's become commonplace components in business infrastructures (Grabski et al, 2012). Despite the many drawbacks reported from implementations gone wrong (Davenport, 1998; Carr, 2003; McAfee and Brynjolfsson, 2008), the market continues to grow with 7% per year (Montgomery, 2012).

During the past couple of years, there have been numerous indications that the market for packaged software is fundamentally changing. With the rise of alternative delivery models such as SaaS and other notions related to cloud computing (Ghormley, 2012; Schenk and Guittard, 2009), industry analysts such as Gartner and Forrester expected the market to experience a radical shift during the end of the first decade in the 21st century (Magnusson et al, 2012). Traditional ERP and CRM models have included on-site installations and substantial configuration initiatives, often making implementations of these systems cumbersome and lengthy. With the rise of a model for packaged software delivered via the Internet, according to a pay-per-view logic of services, the traditional business model of the mega-vendors (SAP, Oracle and Microsoft) was under pressure from smaller, more adaptable players building directly on the emerging technological platform (Magnusson et al, 2012).

According to Weinhardt et al (2009), one of the major trends currently impacting the ERP market is the shift to services and the re-building of traditional ERP vendors into PaaS actors. Instead of controlling the development of functionality in-house, the vendor creates and distributes a development platform and incentive programs for

enticing independent software vendors (ISVs) to create functionality through a process of open innovation (Ku and Cho, 2011; Kim, Kim and Lee, 2010; Nambisan and Sawhney, 2011). Functionality from ISVs is then made available to the PaaS customer, most commonly through some sort of a marketplace. Within packaged software, Salesforce.com's AppExchange was the first of these marketplaces to be acknowledged as successful (Nambisan and Sawhney, 2011; Zittrain, 2009).

2.2 App Stores for Packaged Software Ecosystems

Through an initial review of the available research, we have identified three main dimensions of app stores. The first dimension identified in the literature is related to differences in functional core. Here, the app stores are either intended for the distribution of additional functionality for mobile devices or packaged software. For mobile devices, the level of integration between the app (added functionality) and the devise (functional core) is minimal, primarily focusing on accessing GPS positioning and other hardware related functions. Here, we see examples such as the Apple iTunes Store, Google's Android Market, RIM's BlackBerry App World and Microsoft's Windows marketplace for Mobile (Martens and Hamerman, 2011).

For packaged software, the level of integration between the app (added functionality) and the packaged software (functional core) varies, from configuration packages changing both the business logics layer and the data model, to third-party integration of f.i. Google Maps for viewing data through geo-tagged visualizations (Dilla et al, 2011). With the majority of packaged software solutions being delivered through cloud-based technology, the addition of an app to the software is handled directly on the server side by automated installation. Here, we see examples such as Salesforce.com's AppExchange, Netsuite's SuiteApp and Lawson's Marketplace (Martens and Hamerman, 2011).

In addition to this categorization of the intended "backbone" of the apps (Mobile Device or Packaged software), the scope of control for app stores constitutes the second dimension of categorization. According to Willis et al (2011), app stores may be either private or public. Private app stores are secluded from the general populous, and f.i. gives organizations the possibility of controlling which apps are made available in the corporate environment. The public app stores are not controlled in the same manner, but allow the users full access to all types of apps accepted for publication in the marketplace.

The third dimension refers to the intended customer of the app store. Public app stores for mobile devices such as Apple's App Store and Google's Android Market focus on the consumer (end-user) as a customer. This strategy of consumer-oriented deployment is highlighted as problematic for app stores with packaged software as the functional core. With the functionality often being critical for business and prices being relatively high (when compared to the mobile consumer oriented app stores), user rights related to who may buy apps from a packaged software app store are often restricted (Martens and Hamerman, 2011). Hence, we see a differentiation between app stores directed towards consumers versus corporate decision makers. Table 1 summarizes the dimensions found in the literature.

Table 1. Dimensions and categories of app stores

Dimension	Category 1	Category 2
Functional core	Mobile device	Packaged software
Scope of control	Public	Private
Customer	Consumer	Corporate

2.3 Mechanisms of App Store Governance

There have throughout the years been numerous additions to the development of theory directed towards understanding governance of interorganizational collaborations such as app stores. These collaborations have been investigated through forms such as joint ventures, strategic alliances and business-to-business alliances to name but a few. The rationale behind these studies lies in the reported increase in the importance that these constellations have for business (Ghosal, 1987; Harrigan, 1987; Prahalad and Ramaswamy, 2004). The diverging point between traditional, intra-firm governance and the governance of inter-firm character have been an accentuation through a lower degree of control and an increase of uncertainty (Osborn and Baughn, 1990). This in turns leads to the necessity for alternative takes on governance, with an increased focus on new means for securing control under uncertainty (Harrigan, 1988).

The literature surrounding interorganizational governance encompasses a multitude of different theoretical perspectives such as the Resource Based View, Transaction cost economics and Agency theory (Williamson, 1974). Through a systematic review of the literature (included in the EBSCO database from 1999 to 2012) on interorganizational governance, we have created a composite conceptual framework to aid the future research process. The framework is presented in Figure and Table 1, and is organized into three areas of governance (resource, organization and technology), each with three mechanisms identified in the literature as relevant and important. We will refrain from going into detail about each of the identified mechanisms.

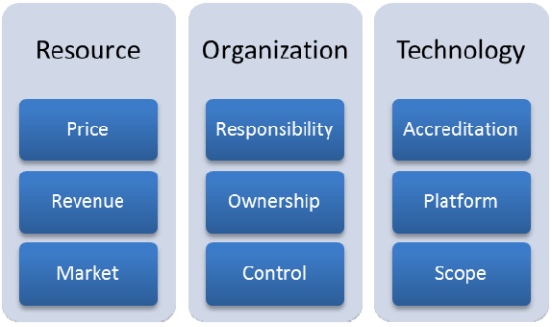


Fig. 1. Proposed tentative framework for App Store governance

Table 2. Description and operationalization of framework

Mechanism	Definition	1	3	5
Price	Price per app	Low mean price (10€)	Medium mean price (500€)	High mean price (1000€)
Revenue	Distribution of revenue among stakeholders	All to developer	Mix	All to seller
Market	Size of the market	Low number of Apps (50)	Medium number of Apps (600)	High number of Apps (1200)
Responsibility	Distribution of responsibility between the stakeholders	Vendor	Developer	Seller
Ownership	Distribution of ownership between the stakeholders	Vendor	Developer	Seller
Control	Bureaucratic control of the marketplace	Trust	Mix	Contracts
Accreditation	Accreditation control over ISVs and other partners who wish to make their functionality available	Trust	Mix	Contracts
Platform	Control over the development platform	No required platform	Mix	Proprietary by vendor
Scope	Size of apps delivered through the marketplace	Small chunks of functionality (internal calculation)	Medium chunks of functionality (Third party integration)	Large chunks of functionality (verticals)

This operationalization will be used as a basis for investigating and describing the different stakeholders perceptions of optimal governance as found in the empirical material.

3 Method

3.1 Empirical Selection

In mid 2011, we were contacted by an ERP vendor concerning a joint research project. The research project would be directed towards understanding the effects of a current innovation initiative that the ERP vendor was working on. This initiative involved the creation of a vendor-owned on-line marketplace for apps, i.e. in this context standardized packages of ERP functionality intended for end-user procurement. In the beginning of 2012 the project was approved funding from a government agency, and the project was initiated in February 2012.

The ERP vendor (SMErp) is the largest ERP vendor in Sweden within the SME segment. The firm has existed since 1992 and has an installed base of 5.000 customers. SMErp follows the traditional ERP business model with a selection of partner organizations controlling the customer interaction and sales. They have a partner structure where partners become certified by the vendor in order to achieve partner status. There are 10 larger partners, constituting 85% of the complete SMErp market. In addition to this, there are 10 smaller partners, constituting the remaining 15% of the market.

For the empirical selection, the research project initiated discussions with informal representatives from the various stakeholders to identify potential interview respondents. This resulted in a list of 20 individuals (7 SMErp, 10 Partner, 3 Customers).

3.2 Data Collection

Throughout the spring of 2012, a series of 20 interviews were conducted with representatives from the vendor's ecosystem. With the change identified as impacting both partners, potential IPOs and customers, these were also included as respondents. The interviews were of a semi-unstructured character, resting solely on three questions (previous, current and future states) and a short introduction by the researcher. Following inspiration from Silverman (2010) and McCracken (1988), the researchers refrained from promoting terminology that may have been construed as value laden throughout the interview. Instead, the respondent was left relatively uninterrupted but for interruptions from the researchers asking the respondent to clarify or exemplify.

All interviews were sound-recorded and transcribed, and handled with the appropriate confidentiality. Provided the delicate nature of the new initiative, we were adamant in establishing the independence of the research group from all involved stakeholders. Provided that one of the intended outputs from the research project was specified as design input for the configuration of the new business model, this required substantial planning.

In parallel with the interviews, the research group gathered documentation related to the current business model and initiative. This included minutes from internal meetings, presentation material from partner gatherings and project documentation.

3.3 Methods of Analysis

Using inspiration from Sarker et al (2012), the analysis involved focusing on the dimension of governance mechanisms in relation to value cocreation. Governance mechanisms were operationalized in line with Sarker et al (ibid), and expanded to encompass aspects commonly related to the field of management accounting. The rationale behind this was to infer additional aspects that could aid in answering of the research question.

Having established the theoretical framework (see section 2), all interviews were read and re-read by the research team in order to categorize the responses into the analytical categories. In addition to the inductively created categories, particularly interesting aspects found in the interviews were discussed within the research team, leading to a further development of the theoretical framework.

4 Results

Given the limitations in size for this publication, we have chosen to present our results in a summarized form in three parts. First, the view of the vendor is presented, followed by the view of the partners. This is in turn followed by a brief gap analysis of the differences in perception between the two stakeholders.

4.1 SMErp's View of Optimal Configuration and Governance

Table 3. Vendor view of optimal configuration and governance

Mechanism	Value	Description
Price	1	The price per app should be low in order to decrease price-related barriers for new functionality. In addition, the cost for building the app is taken by partners, making it difficult to argue for a large revenue-share from the app to the vendor. Since low or no revenue goes to the vendor, a low price is not difficult to accept since it provides value to the customers.
Revenue	3	Revenue should be shared fairly between the developer- and selling partners of the app. Depending on the uniqueness of the knowledge embodied in the app, a larger portion may naturally belong the developing partner.
Market	5	The market should be as big as possible with a large number of apps available. This will build a critical mass and attract new customers.
Responsibility	3	Responsibility over the apps should reside with the developing partner.
Ownership	5	Ownership of the app should reside with the developing partner.
Control	5	Bureaucratic control should be mixed, with a combination of contracts and cultural control through trust.
Accreditation	3	Mixed feelings from the vendor, good with many app-developing partners, also good with high quality-assured work.
Platform	5	The platform for building and publishing apps is the heart of the system and must be kept under strict control by the vendor.
Scope	4	The apps should contain large amounts of packaged industrial knowledge and provide a significant shortcut to non-trivial functionality for the customer

4.2 The Partners' View of Optimal Configuration and Governance

Table 4. Partners' view of optimal configuration and governance

Mechanism	Value	Description
Price	3	The price per app should be in correlation with the effort of producing it and the scarcity if industrial knowledge packaged in the app.
Revenue	3	Revenue should be shared fairly between seller and developer.
Market	2	The market should be limited since only a certain type of functionality is suited to be distributed in this form (see Scope)
Responsibility	4	The responsibility of an app should always reside with the seller. Regardless of who built the app, it must be "washed" and adapted by seller before it can be implemented and used by the customer
Ownership	1	Ownership of the app should follow the responsibility and reside with the seller.
Control	1	The marketplace benefits from a low degree of bureaucratic control, free for the partners to use according to business-opportunities and commercial arrangements and partnerships.
Accreditation	5	In order to contribute to the marketplace, there should be high requirements for accreditation. This will ensure high quality and absence of "ugly hacks" done by less serious actors.
Platform	4	It is important to have a strong commitment and ownership of the platform in order to ensure a continuous development and expansion of functionality. This responsibility is a part of the overall ERP platform strategy and naturally belongs to the vendor.
Scope	2	Partners will use the platform as a basis to systematize and structure all types of customizations. However, they will only package and distribute smaller customizations in the public marketplace. This may change if revenue, ownership and responsibility aspects are more inline with their preference.

4.3 Diverging and Converging Interests in the Paas APP Store

As seen in Figure 2, there are several aspects of governance that may be seen as problematic due to little or no convergence between vendor- and partner preferences. Market, Ownership and Control diverge between optimal configurations for the investigated stakeholders in the software ecosystem. In addition to this, revenue distribution was not regarded as problematic, with both the vendor and partner organizations sharing the same ideas about optimal configuration. We will address these findings in more detail in the section below.

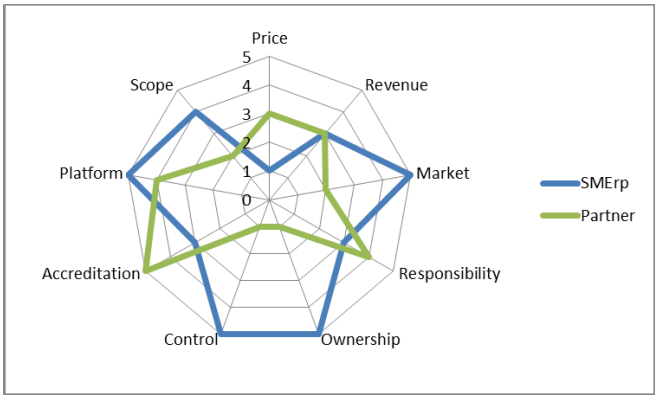


Fig. 2. Differences between vendor and partner perceptions

5 Discussion and Implication

In this study, we highlight three implications related to research and two implications related to practice.

5.1 Implications for Research

The first research implication relates to the methodology applied when developing the model for app store governance. Despite the work still being in an early stage, we believe that the methodology may be used as inspiration when furthering the knowledge of visualization and operationalization that was started by Sarker et al (2012). We believe a continuation in this direction will further the understanding of app store governance for packaged software ecosystems.

Our second implication relates to the value cocreation and distribution within the ecosystems as such. Our study illustrates a gap between vendor, partner and end user that showing a sub-optimization of the platform. Further research towards open source, and open innovation processes (Chesbrough, 2003) may lead to insightful insights and a further understanding of this gap and how it affects IT-governance of the platform.

The third implication for research relates to the overall standardization maturity of the ecosystem, the willingness to join up and collaborate under a predefined set of rules and regulations in order to reach mutual success. As software is increasingly commoditized, also the service offerings surrounding the IT artifact follow the same path of evolvement and development (Davenport, 2005; and Osterwalder, 2011). We believe that the commoditization of services surrounding IT-driven platforms is a future context for study in order to further the understanding of platform governance.

5.2 Implications for Practice

The first implication directed towards practitioners is the identified areas of conflict between the vendor and the partners. The identified main areas are Market, Responsibility and Control. These areas should be given substantial attention in the governance of platform ecosystems. They are all examples of areas where potential conflicts may arise if the governance structures are not adequately place. Looking at these three areas, we consider Responsibility to be the most pressing area. Packed software comes from a tradition of long lifecycles where it is easy to identify responsible stakeholder. This tradition is being challenged in the core by IT driven platforms enabling looser relationship between developers, market makers, integrators and end users. As the complexity of the apps increase, so do their potential value, but also the necessity as such from the software, towards system critical functions and features. The ERP app store owner must make it exceptionally clear who bears what responsibility for app-code in use.

The second implication for practice is acceptance and awareness of the dual market; one the one hand, the market is the Apps offered by the ERP partners towards users of the ERP system. In this case, it is packaged code following an accepted standard that has been developed and offered under market the positive influence of open market competition. This warrants a high quality, adoption and relevance of the attractive apps and a natural exclusion of the not so popular. The secondary market that we have found in our study is the inter- and intra partner market that comes from packaging added functionality and configurations into apps, ready for distribution within and between the partners themselves. In a first wave, this involves the increased level of knowledge sharing among and between partners. In the second wave, one that is not yet reached, it creates the necessary prerequisites of a potential ecosystem of open innovation for the partner network. New ideas could be quickly diffused and the network of certified consultants, through bidding and bartering, could handle calls for new innovations.

6 Conclusions

According to the findings of this study, PaaS ecosystem stakeholders display both diverging and converging ideas about optimal governance of app stores. While they see the distribution of revenue as rather un-problematic and do not differ in their views of what the optimal governance is, issues related to ownership, market size and control are more problematic. Here, the interests are seen to diverge, making them potential areas of conflicts that need to be addressed in the configuration of the app store governance. This study contributes to the emerging field of research on app stores for packaged software through both the development of an operationalization of platform governance and an empirical identification of which aspects of said governance could be seen as sources of potential conflict.

6.1 Directions for Future Research

We see two future projects as viable given this first brief excursion into our case and material. First, a further focus on cocreation of value in platform ecosystems. This project would involve additional analysis of our conducted interviews, using inspiration from Sarker et al (2012) and their study on cocreation of value within packaged software ecosystems. Second, a further focus on the diffusion of ideas, tracing the app store initiative throughout the ecosystem inspiered by (Czarniawska and Sevon, 2005) and (Lounsbury, 2008).

Acknowledgements. We wish to thank Vinnova and the Torsten Söderberg foundation for the monetary support necessary for conducting this research. An alternative version of this paper has recently been accepted for publication in the International Journal of Business Information Systems.

References

1. Böhm, M., Leimeister, S., Riedl, C., Krcmar, H.: Cloud computing: outsourcing 2.0 or a new business model for IT provisioning? Application Management 1, 31–56 (2011)
2. Burkhard, C., Widjaja, T., Buxmann, P.: Software Ecosystems. Wirtschaftsinformatik 54(1), 43–47 (2012)
3. Carr, N.: IT doesn't matter. Harvard Business Review (May 2003)
4. Chesbrough, H.W.: The era of open innovation. MIT Sloan Management Review 44(3), 34–41 (2003)
5. Czarniawska, B., Sevon, G.: Global ideas. How Ideas, Objects and Practices Travel in the Global Economy. Liber & Copenhagen Business School Press (2005)
6. Davenport, T.: The Coming Commoditization of Processes. Hardvard Business Review (June 2005)
7. Davenport, T.H.: Putting the enterprise into the enterprise system. Harvard Business Review 76(4), 121–131 (1998)
8. Dearden (1963)
9. Dilla, W., Janvrin, D.J., Raschke, R.: Interactive DataVisualization: New Directions for Accounting Information Systems Research. Journal of Information Systems 24(2), 1–37 (2010)
10. Gawer, A., Cusumano, M.: Platform leadership: how Intel, Microsoft, and Cisco drive industry innovation. Harvard Business School Press (2002)
11. Ghormley, Y.: Two's a company, three's a cloud: challenges to implementing service models. Journal of Service Science 5(1), 19–28 (2012)
12. Ghosal, S., Bartlett, C.: Creation, Adoption, and Diffusion of Innovations by Subsidiaries of Multinational Corporations. Journal of International Business Studies 19(3), 365–388 (1987)
13. Grabski, S.V., Leech, S.A., Schmidt, P.J.: A review of ERP Research: A future agenda for Accounting Information Systems. Journal of Information Systems 25(1), 37–78 (2011)
14. Harrigan, K.R.: Strategic Alliances: Their New Role in Global Competition. Columbia Journal of World Business 22(2), 67–70 (1987)
15. Harrigan, K.R.: Joint ventures and competitive strategy. Strategic Management Journal (1988)

16. Kim, H.J., Kim, I., Lee, H.G.: The success factors for app store like platform businesses fro the perspective of third-party developers: an empirical study based on a dual model framework. In: PACIS Conference Proceedings (2010)
17. Ku, S.-W., Cho, D.-S.: Platform strategy: an empirical study of the determinants of platform selection of application developers. Journal of International Business and Economy 12(1), 123–143 (2011)
18. Lounsbury, M.: Institutional Rationality and Practice Variation: New Directions in the Institutional Analysis of Practice. Accounting, Organizations and Society 33(4-5), 349–361 (2008)
19. Magnusson, J., Enquist, H., Juell-Skielse, G., Uppström, E.: Incumbents and challengers: conflicting institutional logics in SaaS ERP business models. Journal of Service Science and Management 5(2), 12–25 (2012)
20. Martens, C., Hamerman, P.D.: App stores: a new way to try and buy ERP. Forrester Research (2011)
21. Nambisan, S., Sawhney, M.: Orchestration Processes in Network-Centric Innovation: Evidence From the Field. Academy of Management Perspectives (August 2011)
22. McAfee, A., Brynjolfsson, E.: Investing in the IT That Makes a Competitive Difference. Harvard Business Online (2008)
23. McCracken, G.: The Long interview. Qualitative research methods series, vol. 13. Sage University Paper (1988)
24. Montgomery, N.: Best Practices in ERP Innovation: Toys for the Board, Essential Tools for the Disabled. Gartner Inc report (2012)
25. Nambisan, S., Sawhney, M.: Orchestration processes in network centric innocation: evidence from the field. Academy of Management Perpectives, 40–57 (August 2011)
26. Osborn, R.N., Baughn, C.: Forms of Interorganizational Governance for Multinational Alliances. The Academy of Management Journal 33(3), 503–519 (1990)
27. Prahalad, C.K., Ramaswamy, V.: Co-Creation Experiences: The Next Practice in Value Creation. Journal of Interactive Marketing 18(3), 5–14 (2004)
28. Sarker, et al.: (2012)
29. Schenk, E., Guittard, C.: Crowdsourcing: what can be outsourced to the crowd, and why? (2009)
30. Silverman, D.: Doing qualitative research, 3rd edn. Sage, Thousand Oaks (2010)
31. Weinhardt, et al.: (2009)
32. Willis, D.A., et al.: Re-imagine IT using insights from symposium's analyst keynote. Gartner Inc report (2011)
33. Zittrain, J.: Law and technology: the end of the generative internet. Communications of the ACM 52(1), 18–20 (2009)

Business Model for Analysis of the University Research and Scientific Collaboration: A Case Study

Nataliya Pankratova[2], Oleksandr Maistrenko[1], and Pavlo Maslianko[1]

[1] Applied Mathematics Faculty, National Technical University of Ukraine "KPI"
{o.maistrenko,p.maslianko}@selab.kpi.ua
[2] Institute for Applied System Analysis,
National Technical University of Ukraine "KPI"
natalidmp@gmail.com

Abstract. A university is an entity in the global educational and research community that collaborates with other entities executing joint programs and participating in common projects. In order to succeed in theses activities the university should have a clear understanding of its research results and ways to find the research partners. This is especially important for the Ukrainian scientific community. This paper presents a business model that is based on the scientific collaboration networks. We apply the business model to the educational scientific complex "Institute for Applied System Analysis". The case study reveals the scientific schools of the institute and presents possible collaborators for the future projects.

Keywords: Business model, research analysis, scientific collaboration, social networks.

1 Introduction

A university is an institution working in the research and education areas, including obtaining new knowledge on various subjects, spreading and applying this knowledge, and granting academic degrees. In this paper, we study only the research aspects of the university, including scientific collaboration, and omit its educational activities.

The research activities of the university are sponsored by the government or funded through participation in various grants and commercial projects. It is often the case that obtaining the funds is easier if the university applies for them together with some other institutions, or sometimes a specific setup of the scientific collaborations is required (e.g., a grant-holder from the European Union for an EU-grant). Therefore, the university has to find a research partner with a common research interest that would like to jointly participate in a research/commercial project. Moreover, the search question is currently important for the Ukrainian scientific society that has still quasi closed nature. Though, the Ukrainian institutions have original and interesting research results, but due

W. Abramowicz (Ed.): BIS 2013, LNBIP 157, pp. 50–61, 2013.

to the historical reasons, they are not very well known in the international scientific community. For achieving the satisfactory results of the partner search, the university should have a clear understanding of its research interests, scientific results, and the scientists that stand behind these results. This identifies two research questions investigated in this paper.

Q1. How to analyze the research results of the university?
Q2. How to find research partners for the scientific collaboration?

We base our investigation on the papers published by the scientists from the university under research. The answer to the both research questions is given in a form of the university business model. This business model utilizes the systems theory [20]. The constituents of the business model are the scientific collaboration networks defined by the publications [23]. The case study of the educational scientific complex "Institute for Applied System Analysis" (IASA)[1] shows the validity of the proposed business model.

The scientific collaboration network using co-authorship or co-citation in the published papers has been discussed in the literature for quite some time [21]. Modern studies look into different aspects of these networks such as their structure [18], and the behavioral patterns of the authors [12]. The researchers also study the effects of the collaboration resulting in increase of citations [10], or the productiveness of the scientists [1,22]. The finding of the possible collaboration partners could be achieved through a directed search in the scientific collaboration network [7,26]. However, existing search mechanisms put some requirements either on the structure of the networks (such as connectedness [7]) or require some additional information from the authors (such as a well-defined profile [9]). We get round these requirements by mining the keywords, and using the keyword-based similarity search for the specific research interest.

This paper is structured as follows. Section 2 defines the core definitions used to analyze the research and scientific collaboration. Section 3 describes the university business model, while Section 4 presents a case study of the IASA business model. Section 5 concludes the paper.

2 University Research and Scientific Collaboration

Typically the scientists report the results of their research by means of published papers. Each scientist has an established *research interests* (area of expertise or neighboring areas). In this paper we consider the research interest to be expressed as a word or a word combination. The existing research interests of some domain are structured in the appropriate scientific classifications [3,2,25]. The research interest can be defined more precisely as a set of *keywords*. Such keywords are often found in the scientific publications, and form a part of the publication's metadata [11]. Apart from that, the keywords can be also provided by the scientist as a description of his/her research.

[1] `http://iasa.kpi.ua/`

The scientific collaboration is established based on the shared research interests. A *scientific collaboration* is a human behavior among two or more scientists that facilitates the sharing of meaning and completion of tasks with respect to a mutually-shared superordinate goal and which takes place in social contexts [23]. In this paper, the scientific collaboration is also regarded on the institutional level (i.e., between institutions), though in reality such collaboration is carried out by the individual scientists. We consider a scientific collaboration between the group of scientists to exist, if there is a publication co-authored by them all.

Existing scientific collaborations in the university determine its *scientific schools*. The scientific school is a group of scientists from the same university running a scientific collaboration for the sufficiently long time, and have achieved distinguishable results from this collaboration [16].

The properties of the scientific collaborations can be studied by means of the complex network, namely the *scientific collaboration network* (SCN) [18]. The SCN is a group of scientists participating in scientific collaborations.

3 University Business Model

3.1 Definition of the Business Model

There are multiple definitions of the business model [29,6]. In this paper we consider a business model to describe the rationale of how an organization creates, delivers, and captures value [19], and use the concept of the business model based on the notion of a system [20]. The *system* $S = (E, R)$ is defined as a structure containing a set of *entities* E and a set of *relations* $R \subseteq E^n$ between these entities. There is a *synergy* effect between the entities that causes new features of the system. Such features are not provided by the set of disjoint entities.

The system, its entities and relations have some *features*. The feature is a pair $f = (k, v)$, where k is a definition of the feature, and v is its value for a given system, entity or relation. The values of k and v are not limited to a particular notion or domain, and depend purely on the system under consideration and the problem being solved.

Based on the features of the system, a *view* on the system S is defined as a transition function $P : (S, F) \to (E, R)$, where F is a set of features. The system is split into entities based on the selected features, and all the entities have pairwise different values of these features.

Finally, the business model can be defined as a system derived from the enterprise. The entities of this system are the views on the enterprise that are defined by the selected features. So, the *business model* $BM(S)$ for the enterprise S is a system, whose entities p_i are the views on the enterprise based on the subset $F_{BM}^{(i)}$ of the enterprise's features F_S:

$$BM(S) = \left(\left\{ p_i = P\left(S, F_{BM}^{(i)}\right) | F_{BM}^{(i)} \subseteq F_S \right\}, R_{BM} \right) \tag{1}$$

The views of the business model are conceptualizations. They only define enterprise's entities and relations between them in a formal way, but don't show their essence. In this paper we use complex networks to implement the views.

3.2 Application of the Business Model to the University

The goal of the university business modeling is to answer the research questions stated in Section 1. They form the requirements for the university business model.

Q1. The business model should present the scientists of the university, results of their research, and their scientific collaborations during the specified period of time. The business model should also identify existing scientific schools of the university and show their main research interests.

Q2. Based on the selected research interests of the university, the business model should present a set of educational and research institutions working in the same area, i.e., having the same or similar research interests. These institutions form a scientific community of the university. There might be institutions among them that already run a scientific collaboration with the university. All the others are possible new contacts for the future collaboration. They define a targeted scientific community of the university.
The business model should show both the institutions and the scientists from the (targeted) scientific community.

These requirements determine the views in the university business model on the following: the university, the scientific community, and the targeted scientific community. In terms of (1), the university business model is (S is the university)

$$\begin{aligned} BM(S) &= (\{p_1, p_2, p_3, p_4, p_5\}, R_{BM}) \\ R_{BM} &= \{(p_i, p_j)\}, (i,j) \in \{(1,4),(1,5),(2,4),(2,3),(3,5),(4,5)\} \end{aligned} \tag{2}$$

Figure 1 shows this business model, its views, and the relations between them. The views are depicted as squashed rectangles, relations are solid lines connecting them. The dashed lines represent an existence of the common research interests between the scientists from the university under study and the scientists/institutions in the scientific community.

The university business model uses the following features to separate its views:

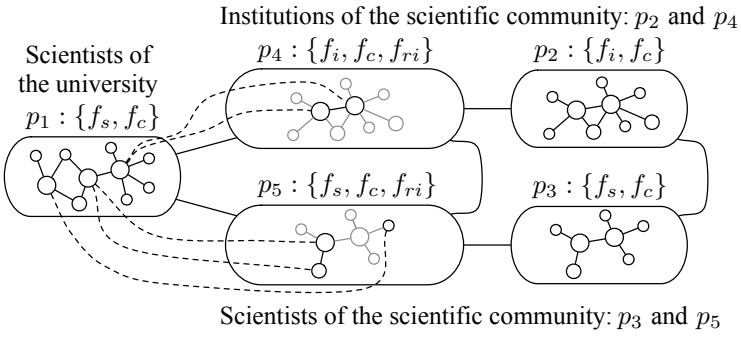


Fig. 1. University business model defined by equation (2)

- f_s with the definition "*scientist*" to determine the individual scientists participating in the scientific collaborations
- f_i with the definition "*institutions*" to determine the individual educational and research institutions participating in the scientific collaborations
- f_c with the definition "*collaboration*" to identify the scientific collaboration between the scientists, and represent them as the relations of the view
- f_{ri} with the definition "*research interest*" to represent the research interest of an entity in the view
- f_l with the definition "*location*" to separate the entities of the university under study from the entities in the (targeted) scientific community

Views. The view $p_1 \equiv P(S, \{f_s, f_c\}) = (E_1, R_1)$ defines the university itself, namely, the scientific collaborations in the university. The entities of this view are the scientists working in the university, and the relations between them represent the collaborations. For all entities in this view, the following relation must hold $\forall e \in E_1 : f_l(e) =$ "university". This view is built to tackle the research question **Q1**.

All the other views in the university business model provide different representations of the scientific community. The view $p_2 \equiv P(S, \{f_i, f_c\}) = (E_2, R_2)$ formalizes the institutions, and the view $p_3 \equiv P(S, \{f_s, f_c\}) = (E_3, R_3)$ shows the scientists in this community. Finally, the views $p_4 \equiv P(S, \{f_i, f_c, f_{ri}\}) = (E_4, R_4)$ and $p_5 \equiv P(S, \{f_s, f_c, f_{ri}\}) = (E_5, R_5)$ narrow down the views p_2 and p_3 correspondingly using the research interests of the university. They define the targeted scientific community. Structurally, these views are derived from the views p_2 and p_3, so that $E_4 \subseteq E_2$ ($R_4 \subseteq R_2$), and $E_5 \subseteq E_3$ ($R_5 \subseteq R_3$). All entities in these four views don't belong to the university, so $\forall e \in E_i : f_l(e) \neq$ "university", where $i = \overline{2,5}$. The views p_2 through p_5 are determined to answer the research question **Q2**.

Each entity in the views p_4 and p_5 must have a similar research interest with at least one entity in the view p_1, so that $\forall e' \in E_i \exists e'' \in E_1 : f_{ri}(e') \approx f_{ri}(e'')$, where $i = \overline{4,5}$. The similarity of the research interest is defined as the *semantic similarity* of the research interests (keywords) [15].

All views of the business model are implemented by SCNs as schematically shown in Figure 1. The greyed out elements in the view p_4 (p_5) show the nodes that are included in the view p_2 (p_3), but not in p_4 (p_5).

Relations. The relations between the view p_1 and the other views are $(p_1, p_i) = \{(e', e'') | e' \in E_1 \wedge e'' \in E_i \wedge f_{ri}(e') \approx f_{ri}(e'')\}$, where $i = \overline{4,5}$.

The relations between the different views on the scientific community $(p_i, p_j) = \{(e', e'') | e' = e'' \wedge e'' \in E_j\}$, where $(i, j) \in \{(2,4), (3,5)\}$.

The relations $(p_l, p_m) = \{(e', e'') | e' \in E_l \wedge e'' \in E_m\}$, where $(l, m) \in \{(2,3), (4,5)\}$, shows that a scientist e'' from one view is affiliated with the institute e' from the corresponding view. Each scientist must be affiliated with at least one institute, and each institute must be represented by at least one scientist.

4 Case Study

The case study has been carried out for the educational scientific complex "Institute for Applied System Analysis" (IASA). The business model (2) provides a guidance for analysis of the IASA research and scientific collaboration. First, we investigate the research at IASA and create the view p_1 of the business model defined by equation (2). Based on this view, we identify scientific schools of IASA, and their research interests. In Section 4.2, we determine the scientific community for IASA and build views p_2 and p_3. In Section 4.3, we allocate the targeted scientific community and create views p_4 and p_5.

4.1 Research at IASA

IASA is the educational scientific complex established in 1997 on the base of Chair of Mathematical Methods of System Analysis (National Technical University of Ukraine "Kiev Polytechnic Institute") and Institute for Applied System Analysis (National Academy of Sciences of Ukraine). The institute works mostly in the field of the applied system analysis based on the mathematical methods (numerical methods, optimization problems, differential equations), and systems design (CAD).

There are 4 faculties (Faculty of Systems Research; Faculty of Pre-Institute Training; Faculty of Second Higher and Post-Diploma Education; and Faculty of Course Training) and 5 scientific departments (Numerical Methods of Optimization; Information Resources; Laboratory of Nonlinear Analysis for Differential-Operator Systems; Mathematical Methods of System Analysis; and Applied Nonlinear Analysis) in IASA. In the following, we have considered only the Faculty of Systems Research and the IASA administration. The list of the employees has been taken from the IASA website. There are 20 employees in the list, and it includes key IASA scientists involved in the educational process.

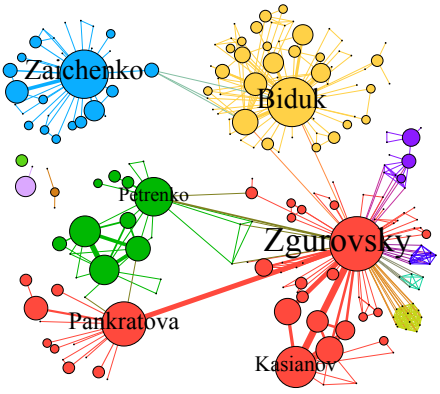


Fig. 2. SCN for the scientists of IASA (view p_1)

For the selected scientists, we produced a list of their publications in the last 10 years, i.e., from 2002 to 2012. This time period was selected to skip the first five years of the IASA existence. The correctness of the choice is confirmed by the fact that the fundamental works on the applied system analysis have been published by the scientists from IASA in 2005–2007 [27,28].

Table 1. The scientific schools of IASA (extracted from p_1)

Colour	Node coverage	Publication count	Key scientists	Research interests
Red	29%	216	Zgurovsky, Pankratova, and Kasianov	"systems analysis", "sustainable development", and "foreseeing"
Yellow	28%	73	Biduk	"risk analysis", "forecasting"
Blue	14%	70	Zaichenko	"financial engineering", "modeling", "optimization", "operations research"
Green	10%	35	Petrenko	"grid", "scientific workflow", "'distributed computing", "'microprocessor systems"

The publication search was conducted for each scientist using the "Publish or Perish" application [4].

Figure 2 presents the implementation of the view p_1. It is based on 424 publications including 206 authors in total (20 of them are the initially selected IASA employees). Figure 2 shows the names only of the key scientists in IASA. The size of the node is proportional to the number of scientific publications of the scientists. The thickness of the edge is determined by its weight, and it is proportional to the number of common publications of the authors connected by the edge.

The search has added another 16 authors affiliated with IASA who are either students, past employees, or work in a different division of IASA. Including them, the average number of publications of an author affiliated with IASA equals to 16.778 (whereas the average number of publications for the complete SCN is 4.218). The average number of the scientific collaborations for the whole SCN corresponds to the average weighted degree, and it is equal to 7.495.

Running the modularity analysis over this SCN [5], we identified four major classes of modularity that represent the scientific schools of IASA (modularity is 0.717). These communities are shown in different colors in the figure, and they cover about 80% of the SCN nodes. Table 1 shows these scientific schools and their research interests.

4.2 Scientific Community for IASA

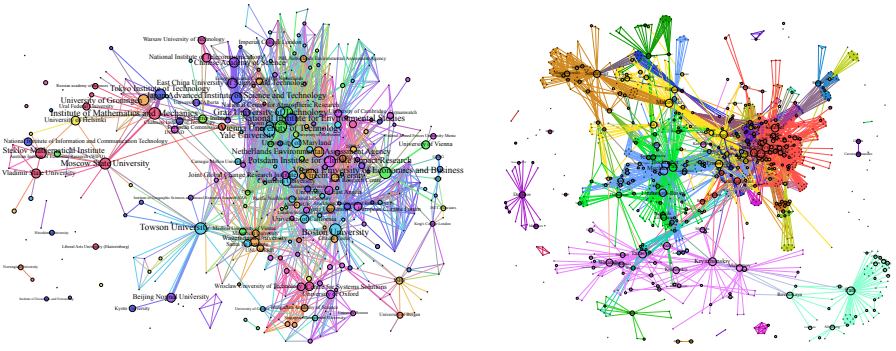
One of the main research interests of IASA is "*applied system analysis*". There are multiple educational and research institutions worldwide that have the same research interest, e.g., International Institute of the Applied Systems Analysis (IIASA)[2], Institute of Systems Analysis of Russian Academy of Science, International Federation for Systems Research, Institut für Systemwissenschaften, Innovations- & Nachhaltigkeitsforschung (University of Graz), etc. After discussions of the IASA business model with the IASA administration, we decided

[2] `http://www.iiasa.ac.at/`

to use a part of the community around IIASA as a scientific community. The scientific community is formed by the scientists participating in the following IIASA research programs: "Advanced Systems Analysis" (including "Dynamic Systems" and "Integrated Modelling Environment"), "Risk, Policy, and Vulnerability", and "Transitions to New Technologies". These research programs highly correlate with the scientific interests of IASA.

Using these research programs of IIASA and time period from 2002 to 2012 (same as in Section 4.1), we performed a publication search through the IIASA website. The search resulted in 875 publications, where 120 publications are the IIASA interim reports. These publications are written by 905 authors working for 362 institutions (excluding IIASA).

This data was used to create the SCNs for the views p_2 and p_3 that are shown in Figure 3 (labels are shown only for the biggest nodes). The size of the node is proportional to the number of publications. It can be bigger than the number of scientific collaborations of this institution (author) that corresponds to the weighted degree of the node. In Figure 3(a) IIASA is not shown, because it is connected with all the nodes in the network and would clatter the visualization. However, it is possible to study the scientists of IIASA using the view p_3.



(a) View p_2: institutions (b) View p_3: authors

Fig. 3. SCNs for the targeted scientific community of IASA

The nodes in Figure 3(a) are colored by the geographical location of the institution. The top five countries (by institution count) are USA, Germany, UK, Netherlands, and Austria. Moreover, 23 countries have more than 1% of the nodes, and these countries cover almost 92% of the institutions. Interestingly enough, Ukraine is one of these countries. The top ten institutions in the graph by the publication count (excluding IIASA) are three local universities from Austria, as well as three institutions from the USA, two from Russia, one from Germany,

and one from Japan. The average number of publications for an institute in this SCN equals to 5.4.

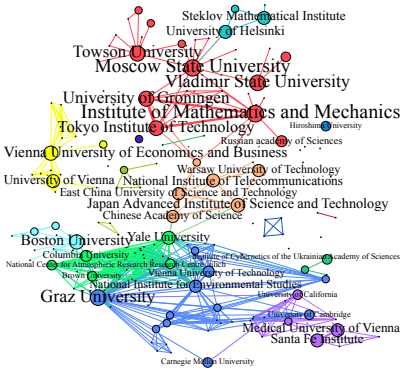
In Figure 3(b) the nodes are colored by the modularity class (modularity is 0.699). The number of communities is 60, and only 11 (19) of them have more than 20 (10) members. The average number of publications for a scientist in this SCN is 3. The view p_3 allows finding the most important scientists in the targeted scientific community, and their collaborators. Obviously, these scientists mostly work for IIASA, due to the selection of the scientific community.

4.3 Scientific Collaboration: IASA and IIASA

To create the views p_4 and p_5, we had to find the likeness of the nodes in views p_2 and p_3 (Section 4.2) and the research interest for each scientific school (Table 1). First, for each institution and author identified in Sections 4.2, we identified their research interests. The research interest is expressed as the keywords of the publications. Therefore, we identified the keywords for the selected publications manually and using the *keyword extraction* tool Maui [17]. Maui analyses the input pdf file and produces a set of the keywords. During the processing, we didn't specify the domain for Maui. The analysis of the selected publications has shown that the keywords are specified only in about 19% of cases (166 publications). Some of the remaining publications could be accessed through the web. We used Maui to extract their keywords [17]. The tool provided the keywords for another 29% of the selected publications (257 publications). So, in total, we could identify the keywords for about 50% of the selected publications.

As a measure of the likeness between the scientific school of IASA and a node in the selected scientific community, we chose a *semantic similarity* [15] of the research interest and the keywords of the publication. There are different approaches for computation of this measure [8,13,24]. We had to take into consideration that a keyword can be a word combination. Therefore, we had to compute the semantic similarity of the phrases, whereas most of the tools are limited to the individual words.

We have chosen the Google Similarity Distance (GSD) for the computations [8]. The GSD requires a number of the documents that are searched by Google. This number has been estimated using [14], and it is equal to around 40 billion pages as of October, 2012. The similarity between the research interest and the publication is the maximal GSD of the research interest and

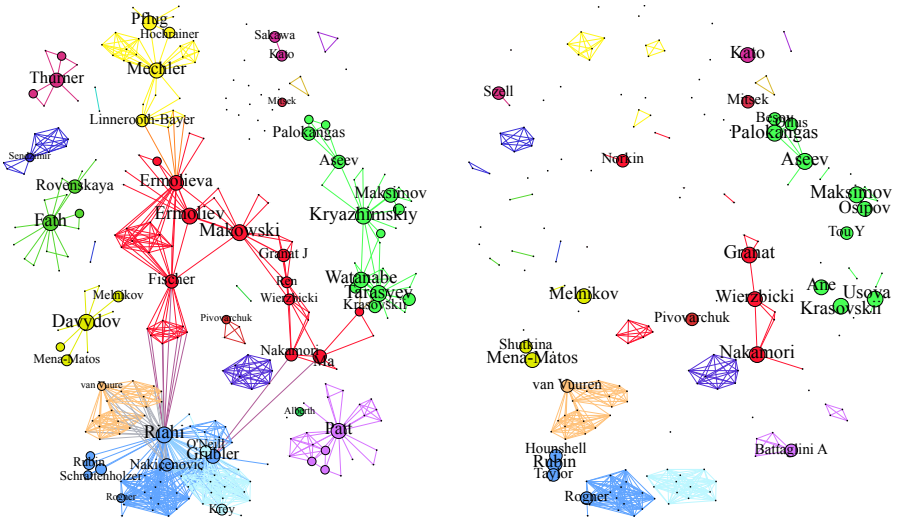


Number of nodes = 149

Fig. 4. View p_4 (institutions) for the targeted scientific community of IASA filtered for the research interest "*sustainable development*"

the keywords. If the keywords were not determined, we have taken the GSD of the research interest and the title of the publication. The results of the GSD computations have been normalized. Finally, the publications have been filtered with a threshold to leave only the publications that share the research interests with IASA to some selected extent. The threshold selection was a compromise between the number of authors/institutions in the views and the similarity of the publications with the research interests. After some experiments, the filtering has been performed with a threshold of 0.5, and has resulted in 274 authors (29% of the authors in p_3) working for 149 institutions (40% of the institutions in p_2).

Figure 4 and Figure 5(a) show the SCNs for the views p_4 and p_5 for the research interest "sustainable development". This research interest corresponds to the red scientific school of IASA (Table 1). We built similar views for all scientific interests of the IASA scientific schools, but we omit them in this paper.



Number of nodes = 274
(a) With IIASA authors

Number of nodes = 190
(b) Without IIASA authors

Fig. 5. View p_5 (authors) for the targeted scientific community of IASA filtered for the research interest "*sustainable development*"

In Figure 4, the nodes are colored by the modularity classes (modularity is 0.643). There are 28 communities, and 7 (12) of them contain more than 5 (2) nodes. For the view p_4, we have selected the top 20 institutions by the publication count. They include 4 universities from Austria and IIASA, 4 universities from the USA, and 4 from Russia. The names of the biggest institutions by the publication count are shown in Figure 4. In such a case, the tendency for the

collaboration with local researchers that was studied in [12] becomes apparent. It can be tracked for some institutions from Russia, China, and Austria.

The nodes in Figure 5(a) are colored by the modularity classes (modularity is 0.756). There are 37 communities, and 12 of them contain more than 5 nodes. As compared to the view p_3, p_5 has 5 communities with more than 20 nodes. For the view p_5, we have identified the key researchers for the biggest modularity classes and presented them to the IASA administration as possible research collaborators in the sustainable development. Their names are shown in Figure 5(a). As noticed for the view p_3, the key researchers are the IIASA employees. This fact is explained by selection of the targeted scientific community. Therefore, we created another visualization of p_5 by removing IIASA employees as shown in Figure 5(b). Some of the scientists are in both figures emphasizing their contribution to the targeted scientific community.

5 Conclusion

This paper proposes the university business model reflecting the research aspects of the university. This business model demonstrates the application of the scientific collaboration networks to the business modeling domain. Such business model allows university to find new collaborators in the specified scientific community. The business model consists of the several complex networks representing scientific collaboration both inside the university and between other institutions in the selected research area of the university. These parts are combined together to discover new properties of the scientific collaboration networks. A case study has been completed for the educational scientific complex "Institute for Applied System Analysis" using this business model. We have started with the analysis of the IASA scientific schools and as a result obtained a list of the possible collaborators in the selected research areas. The outlooks for this work include development of the more precise paper search/selection mechanism, and an automation of the publications' metadata processing.

References

1. Abbasi, A., Altmann, J.: On the correlation between research performance and social network analysis measures applied to research collaboration networks. In: Proceedings of the HICSS 2011, pp. 1–10. IEEE (2011)
2. ACM: The ACM Computing Classification System (2012)
3. AMS: Mathematics Subject Classification (2010)
4. Bensman, S.: Anne-Wil Harzing: The publish or perish book: Your guide to effective and responsible citation analysis. Scientometrics 88, 339–342 (2011)
5. Blondel, V.D., Guillaume, J.L., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiment 10, 8 (2008)
6. Burkhart, T., Krumeich, J., Werth, D., Loos, P.: Analyzing the business model concept — a comprehensive classification of literature. In: Proceedings of the ICIS 2011. AIS (2011)

7. Chirita, P.A., Damian, A., Nejdl, W., Siberski, W.: Search strategies for scientific collaboration networks. In: Proceedings of the 2005 ACM Workshop on Information Retrieval in Peer-to-Peer Networks, P2PIR 2005, pp. 33–40. ACM (2005)
8. Cilibrasi, R., Vitanyi, P.: The google similarity distance. IEEE Transactions on Knowledge and Data Engineering 19(3), 370–383 (2007)
9. Diederich, J., Iofciu, T.: Finding communities of practice from user profiles based on folksonomies. In: Proceedings of the EC-TEL 2006 Workshops. CEUR Workshop Proceedings, vol. 213, pp. 288–297. CEUR-WS.org (2006)
10. Divakarmurthy, P., Menezes, R.: The effect of citations to collaboration networks. In: Menezes, R., Evsukoff, A., González, M.C. (eds.) Complex Networks. SCI, vol. 424, pp. 177–185. Springer, Heidelberg (2013)
11. Dushay, N., Hillmann, D.I.: Analyzing metadata for effective use and re-use. In: Proc. of the Dublin Core Metadata Conference 2003, pp. 17:1–17:10. DCMI (2003)
12. Evans, T., Lambiotte, R., Panzarasa, P.: Community structure and patterns of scientific collaboration in business and management. Scientometrics 89, 381–396 (2011)
13. Gabrilovich, E., Markovitch, S.: Computing semantic relatedness using wikipedia-based explicit semantic analysis. In: Proceedings of the IJCAI 2007, pp. 1606–1611. AAAI (2007)
14. de Kunder, M.: The size of the world wide web, `http://www.worldwidewebsize.com`
15. Lin, D.: An information-theoretic definition of similarity. In: Proceedings of the 15th International Conf. on Machine Learning, pp. 296–304. Morgan Kaufmann (1998)
16. Malciené, L.: Scientometric analysis of a scientific school. Scientometrics 15, 73–85 (1989)
17. Medelyan, O., Frank, E., Witten, I.H.: Human-competitive tagging using automatic keyphrase extraction. In: Proc. of the EMNLP 2009, pp. 1318–1327. ACL (2009)
18. Newman, M.E.J.: The structure of scientific collaboration networks. Proceedings of the National Academy of Sciences 98(2), 404–409 (2001)
19. Osterwalder, A., Pigneur, Y.: Business Model Generation: A Handbook for Visionaries, Game Changers, and Challengers, 1st edn. Wiley (2010)
20. Pankratova, N., Maistrenko, O., Maslianko, P.: System definition of the business/enterprise model. In: Herrero, P., Panetto, H., Meersman, R., Dillon, T. (eds.) OTM 2012 Workshops. LNCS, vol. 7567, pp. 134–143. Springer, Heidelberg (2012)
21. de Solla Price, D.J.: Networks of scientific papers. Science 149, 510–515 (1965)
22. Reijers, H.A., Song, M., Romero, H., Dayal, U., Eder, J., Koehler, J.: A Collaboration and Productiveness Analysis of the BPM Community. In: Dayal, U., Eder, J., Koehler, J., Reijers, H.A. (eds.) BPM 2009. LNCS, vol. 5701, pp. 1–14. Springer, Heidelberg (2009)
23. Sonnenwald, D.H.: Scientific collaboration. Annual Review of Information Science and Technology 41(1), 643–681 (2007)
24. Tsatsaronis, G., Varlamis, I., Vazirgiannis, M.: Text relatedness based on a word thesaurus. Journal of Artificial Intelligence Research 37, 1–39 (2010)
25. UDC Consortium: Universal Decimal Classification
26. Yu, B., Singh, M.P.: Searching social networks. In: AAMAS, pp. 65–72. ACM (2003)
27. Zgurovsky, M.Z., Pankratova, N.D.: System Analysis: Problems, methodology, applications. "Naukova Dumka" Kyiv (2005) (published in Russian)
28. Zgurovsky, M.Z., Pankratova, N.D.: System Analysis: Theory and Applications. Springer (2007)
29. Zott, C., Amit, R., Massa, L.: The business model: Recent developments and future research. Journal of Management 37(4), 1019–1042 (2011)

Business Process Model Overview: Determining the Capability of a Process Model Using Ontologies*

Wassim Derguech and Sami Bhiri

National University of Ireland, Galway - Digital Enterprise Research Institute
firstname.lastname@deri.org
www.deri.org

Abstract. Representing business process models to stakeholders depends on the required level of details. While technical team is interested in HOW processes are performed and their detailed representation, strategic management team is more concerned by WHAT processes perform by featuring business properties. In order to allow for a seamless navigation between the WHAT and the HOW levels of process representations, we propose in this paper an algorithm that takes as input a detailed process model and provides a quick overview of its capability. Our solution imposes that the elements of the input model are annotated with their capabilities. In this paper we reuse a previous work on capability modeling.

Keywords: Capability, Composition, Control Flow, Aggregation, Abstraction.

1 Introduction

Process Aware Information Systems [1] allow to manage and execute business processes involving several components on the basis of process models. These models constitute a central element that is being shared among various stakeholders.

A business process model can detail various elements: activities, data objects, control flow, etc. Therefore, not all the stakeholders are interested in all these details; e.g., the strategic management team is more interested in the WHAT is being performed, however, the technical team is interested in HOW tasks are performed. Consequently, there is a lot of effort put towards finding the right details that need to be presented to the involved stakeholders. For example, when it comes to privacy concerns when presenting processes to business partners, [2] suggests hiding unwanted process elements while preserving the entire process consistency, whereas [3] presents an approach that allows for a customized representation of process models with respect to the user preferences, while [4] proposes to simply reduce the complexity of process models.

Whether the aim is hiding details for privacy reasons, providing different process views or reducing the complexity of models, the object remains the same: transforming process models from most to least detailed ones.

Business Process Models Abstraction (BPMA for short) is a promising technique that allows for a seamless navigation from a detailed process model to an abstract one. Two strategies can be used in BPMA. The first one consists of leaving out unwanted

* This work is funded by the Lion II project supported by Science Foundation Ireland under grant number SFI/08/CE/I1380 Lion-2.

W. Abramowicz (Ed.): BIS 2013, LNBIP 157, pp. 62–74, 2013.

components of the model [2]. Users can for example visualize only the elements they are interested in. With such solution, only essential elements are kept, however, the entire overview of the process model is partially presented. The second one consists of aggregating several components into a single abstract one [3–8]. For example, the activities "book flight" and "book hotel" can be aggregated into an abstract activity "arrange trip". Consequently, an entire process model can be represented at several levels of abstraction. It can even be abstracted into a single activity. In such settings, current solutions limit the result of aggregation to a single label which gives a shallow representation of the capability of the entire process.

We argue that a single label does not carry enough information to adequately describe the semantics of the functionality of an entire process model. Fig. 1 depicts an example containing a process model for the examination procedure of an importation process[1] This example would be abstracted into one activity that will be presented by a single label (e.g., Examination of cargo) using the approach proposed in [8].

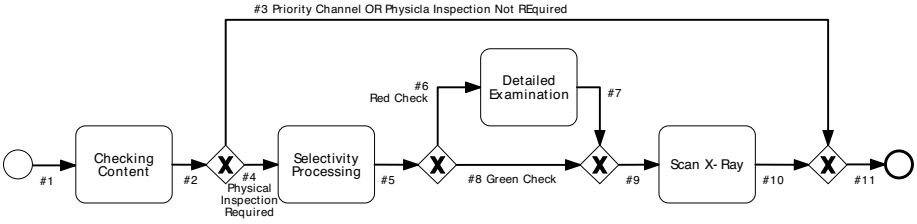


Fig. 1. Examination of cargo procedure at Davao City seaport in Philippines

We propose in this paper another technique that allows moving from an entire process model to its functional description by aggregating all the capabilities of the process elements into a single composed capability. A capability should feature functional domain properties and not limited to a single label. The capability of the process model depicted in Fig. 1 should report that after checking the content of the cargo a decision about physical inspection is made. If the cargo goes through a priority channel or if a physical inspection is not required, then it is directly released without inspection, otherwise a physical inspection is required. In this case, a red or green check is performed. A red check goes through a detailed examination of goods and an X-ray scan, however, a green check needs only an X-ray scan.

The contribution of this paper is an algorithm that computes the capability of an entire process model having as input an annotated model. We use the model in Fig. 1 as a running example. The algorithm will be presented in Section 3. By annotated model we mean a control flow where each activity is annotated by its capability considering the conceptual model defined in a previous work [9]. In Section 2 we recall this conceptual model and introduce the notation that we use for an annotated process model. Before concluding the paper in Section 5, we review important contributions related to our work in Section 4.

[1] This model is available at `http://kjri-davao.com/?page=news&siteLanguage=English&add ress%20link=127&cat=Economics` as accessed on 03-09-2012.

2 Background

2.1 A Meta-model for Describing Capabilities

A capability denotes what an action does either in terms of world effects or returned information [2] . In the literature, we can distinguish three families of approaches that tackled the problem of capability modelling either directly or indirectly. The first family includes semantic Web services models (WSMO [10] and OWL-S [11]) which model capabilities as Input, Output, Preconditions and Effects (IOPE). IOPEs do not represent explicitly domain features and their interpretation is heavily dependent on reasoning.

The second family of related efforts concerns semantic annotations of invocation interfaces (SA-WSDL and SA-REST) [12, 13]. Such approaches attempt to provide alternative solutions to top-down semantic approaches (WSMO and OWL-S) by starting from existing descriptions such as WSDL and annotate them with semantic information. These approaches are describing invocation interfaces rather than concrete capabilities.

The third family includes frame-based approaches for modelling capabilities. Oaks et al., [14] give a nice overview of related approaches and propose a model for describing service capabilities as such. The proposed model distinguishes in particular the corresponding action verb and informational attributes in addition to the classical IOPE. In such work, the semantics of capabilities remain defined via the IOPE paradigm and therefore has the same problems as the first family of approaches described above.

All of the previously discussed approaches describe capabilities without featuring functional domain properties. A capability is highly tight to its implementation (i.e, invocation interface) or related to the description of another concept (i.e., services). We strongly support the idea of considering the capability as independent concept that describes what a program, a business process, a service, etc. does from a functional perspective. A capability should not be limited only to single label or action verb but also should consider a proper description of functional domain related properties.

In this context, we propose in Definition 1, a capability meta model featuring functional domain properties via an "*ActionVerb*" and a set of "*Attribute*"and "*Value*" pairs.

Definition 1 (Capability) *A tuple* Cap = (ActionVerb,Attributes) *is a capability, where:*

- ActionVerb*: We consider the action verb as a concept from an actions ontology.*
- Attributes*: Represents a set of pairs* (AttributeName, AttributeValue) *that correspond to a set of functional characteristics. An* AttributeName *corresponds to the identifier of the attribute and* AttributeValue *corresponds to its value.*

We define action verbs with respect to an action verb schema (http://vocab.deri.ie/av) where we define the concept *ActionVerb* as an *rdfs:subClassOf skos:Concept*. The proposed schema defines for this concept three properties: *av:hasPart* and av:hasOptionalPart which are used to build a hierarchy of action verbs expressing meronymy relations between them. A meronymy relation holds between two concepts if one of them is part of the other. We call this hierarchy as *Actions Ontology*. Examples

[2] OASIS Reference Model for Service Oriented Architecture 1.0, `http://www.oasis-open.org/committees/download.php/19679/soa-rm-cs.pdf`

of Action Ontologies include the MIT process handbook [15] is a good example of an actions ontology that contains over 5,000 activities organized using various relations including meronymy. In this work, we created an actions ontology (http://vocab.deri.ie/imp) for the import procedure in custom clearance domain. We will discuss in detail this actions ontology (see Fig. 2) later in this paper.

Attributes also are defined in a domain ontology that allows to describe capabilities properly. The value of each attribute is described according to our Capability Meta Model (http://vocab.deri.ie/cap#) presented in detail in a previous work [9] where we define the different types a value can have including *EnumerationValue*, *DynamicValue*, *ConditionalValue*, etc.

Listing 1.1. Selectivity Processing Description

```
:Phil_SelectivityProcessing a bp:AtomicTask;
bp:hasCapability :Phil_Cap_SelectivityProcessing.

:Phil_Cap_SelectivityProcessing a cap:Capability;
cap:hasActionVerb imp:SelectivityProcessing;
impc:hasCargo :Phil_Cargo;
impc:hasTypeOfCheck :Phil_TypeOfCheck.

:Phil_TypeOfCheck a impc:TypeOfCheck, cap:EnumerationValue;
cap:hasElement impc:RedCheck;
cap:hasElement impc:GreenCheck.
```

Listing 1.1 an example describing the task "Selectivity Processing" from the process model depicted in Fig. 1. This task has as identifier : $Phil\_SelectivityProcessing$, is declared as an atomic task and has a capability : $Phil\_Cap\_SelectivityProcessing$. This capability has as action verb imp : $SelectivityProcessing$ and two attributes : $Phil\_Cargo$ and : $Phil\_TypeOfCheck$. The second attribute is defined as cap : $EnumerationValue$ which means it has more than one option which are $impc$: $RedCheck$ and $impc$: $GreenCheck$. We use in this Listing several namespaces such as $impc$ for referring to the Import Process Capabilities Ontology, imp for the Actions Ontology of the import domain(http://vocab.deri.ie/imp), cap for our Capability Meta Model [5] and bp for our Business Process Vocabulary (http://vocab.deri.ie/bp#).

An advantage of our model is that it exposes the capability as machine processable and end-user centric. Indeed, making functional properties explicit and distinct makes their manipulation and interpretation very easy either by a machine or human. Whereas, within the IOPE paradigm, Preconditions and Effects are expressed via logical formulas that require reasoning and further processing for making them human understandable.

In this paper, we are interested in composing capabilities and we aim to provide an algorithm that given a capability annotated control flow (i.e., a business process or subprocess) it determines the corresponding capability. In the following, we introduce the concept of capability annotated control flow.

2.2 Annotated Control Flow Model

In this paper, the input of our algorithm is a control flow that has capability annotations. In other words, each activity node in the control flow model is annotated by its capability with respect to Definition 1. In this section, we introduce our notation for describing the concept of Annotated Control Flow Model that is illustrated in Definition 2.

Definition 2 (Annotated Control Flow Model) *An* Annotated Control Flow Model *is a directed graph $G =< N, C, Cap >$, where N is a set of nodes: InitialNode, FinalNode, ActivityNode, ANDsplit, ANDjoin, ORsplit, ORjoin, XORsplit, XORjoin and C is a set of graph connectors. Cap is an annotation function that associates with each activity node n a tuple Cap(n) = (ActionVerb (n), Attributes(n)). In addition:*

- $\forall n \in N$, *$\bullet n/n\bullet$ denotes the set of incoming /outgoing connectors of n.*
- $\forall c \in C$, *if $c \in \bullet n$ then $c \notin n\bullet$ and if $c \in n\bullet$ then $c \notin \bullet n$*
- *for each split node, n: $|\bullet n| = 1$ and $|n \bullet| > 1$;*
- *for each join node, n: $|\bullet n| > 1$ and $|n \bullet| = 1$;*
- *for each activity node n: $|\bullet n| = 1$ and $|n \bullet| = 1$;*
- $|\bullet InitialNode| = 0$ *and* $|InitialNode \bullet| = 1$;
- $|\bullet FinalNode| = 1$ *and* $|FinalNode \bullet| = 0$;
- *if n is an ORsplit or a XORsplit: $\forall c \in n\bullet$, c is guarded by a condition, $cond_c$.*
- *a path $p(n, m) = \{c_0, ..., c_i\}$ is the set of consecutive connectors from a node n to m such that $c_0 \in n\bullet$ and $c_i \in \bullet m$.*
- *for each ORjoin and XORjoin node n: every connector $c \in \bullet n$ is guarded by a condition, $cond_c$.*
- *each node $n \in N$ is on a path from the $InitialNode$ to the $FinalNode$.*
- *for each activity node n: $ActionVerb(n)$ refers to the action verb of n.*
- *for each activity node n: $Attributes(n)$ refers to the set of attributes of n.*
- *for each connector c: $Condition(c)$ refers the the condition $cond_c$ that guards it.*

Additionally, an Annotated Control Flow Model does not contain cycles and is a well structured model. It imposes that each split node (i.e., ANDsplit, ORsplit and XORsplit) has a corresponding join node (i.e., ANDjoin, ORjoin and XORjoin) [16].

3 Capability Composition

As it has been previously mentioned, the input of our algorithm is a capability annotated control flow. We have manually created capabilities for each activity node of the process model depicted in Fig. 1. Table 1 shows these annotations: action verbs and attributes. For presentation purposes, action verbs are same as labels in Fig. 1. Each attribute is described via a name followed by = followed by its value and its category between [ ] (categories will be introduced later in this section). For example the capability of the "Selectivity Processing" task in the model of Fig. 1 has as action verb *Selectivity Processing* and as attributes (i) *cargo* having the value *imp:cargo* and (ii) *TypeOfCheck* as an Enumeration of imp:RedCheck and imp:GreenCheck of category Dominant. The third column contains a textual description of the capability. Please note that this textual description is not mandatory, it is used here only for explanation purposes. This capability is formally presented using RDF in Listing 1.1.

3.1 Determining the ActionVerb of the Composed Capability

The action verb is a madatory attribute in the capability description. Its value is taken from an Actions Ontology that is also used for determining the action verb of composed

Table 1. This table shows the capability annotations of the activities of Fig. 1

Activity	Action verb and Attributes	Textual description
Checking Content	ActionVerb = Checking Content Cargo = imp:cargo [Passive] ExamDecision = {imp:PhysicalInspectionRequired, imp:PhysicalInspectionNotRequired, imp:PriorityChannel}[Dominant]	This activity consists of **checking the content** of the **cargo** in order to take a **decision** about the necessity of a physical check.
Selectivity Processing	ActionVerb = Selectivity Processing Cargo = imp:cargo [Passive] TypeOfCheck = {imp:RedCheck, imp:GreenCheck} [Dominant]	This activity consists of **selecting** the **type of check** that needs to be done.
Detailed Examination	ActionVerb = Detailed Examination Cargo = imp:cargo [Passive] ExamType = imp:detailed [Passive]	This activity consists of performing a **detailed examination** of the **cargo**.
Scan X-Ray	ActionVerb = Scan X-Ray Cargo = imp:cargo [Passive] ExamType = imp:X-Ray [Passive]	This activity consists of performing an **X-Ray scan** of the **cargo**.

capabilities. Actually, a composed capability has as action verb the corresponding lowest common ancestor (LCA) of the action verbs of its components.

In our work, we created an Actions Ontology for Import procedures [4] that is illustrated in Fig. 2. Using this ontology for determining the action verb of the composed capability of the entire process model depicted in Fig. 1 consists of looking for the LCA of all the action verbs of tasks of that process model : *LCA(Checking Content, Selectivity Processing, Detailed Examination, Scan X-Ray) = Examination of Cargo*.

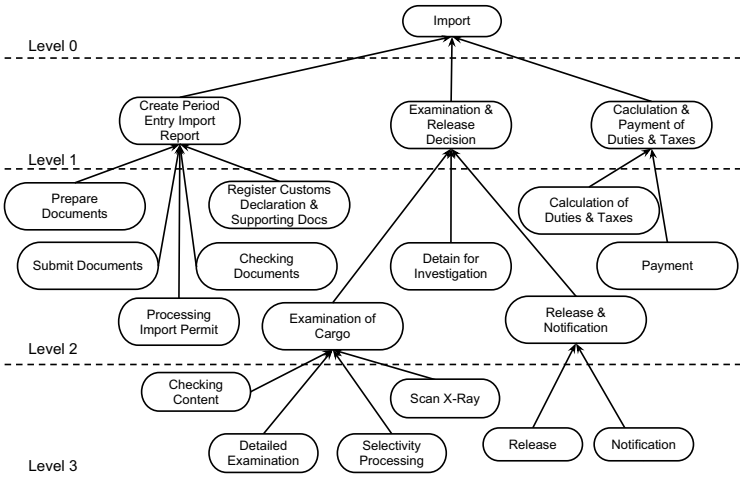


Fig. 2. Actions Ontology of the import domain

Ideally, all the action verbs used in the model are taken from the same actions ontology like in our running example. Modelers can use actions taken from different action ontologies. Instead of considering a single actions ontology, we need to take into account all the possible ontologies used in assigning action verbs to the capabilities of a process model. In such case a more elaborated method as presented in [8] is needed.

Moreover, the use of the LCA for determining the action verb of a composed capability is not always sufficient especially if we do not have the complete set of verbs in the meronymy. This problem can be addressed as follows: if all the actions that are linked with the *av:hasPart* property are present, then we defined the corresponding LCA otherwise the result would the set of all the composing action verbs.

3.2 Determining the Set of Attributes of the Composed Capability

We propose in this paper a propagation algorithm for determining the set of attributes of a composed capability. The idea of our algorithm is to start from the *InitialNode* then firing all the nodes one by one and propagating the subsequent attributes until reaching the *FinalNode*. Each node introduces some changes on the set of attributes. The set of attributes propagated at a particular node is marked on the outgoing connector(s). The Attributes Propagation Algorithm operates as follows:

Attributes Propagation: Let $G =< N, C, Cap >$ be a capability annotated control flow and $\mathcal{A} = \bigcup_{\forall n \in N} Attributes(n)$. We define the function attributes initialization $Att_0 : C \rightarrow \mathcal{A} \bigcup \{\perp\}$ such that $\forall c \in C$:

- $Att_0(c) = \{\}$ if $c = InitialNode\bullet$
- $Att_0(c) = \{\perp\}$ otherwise (the symbol $\{\perp\}$ means that the value is unknown)

Let the two functions $Att_k, Att_{k+1} : C \rightarrow \mathcal{A} \bigcup \{\perp\}$. $\forall n \in N : Att_{k+1}$ is the propagation of Att_k at the node n iff: $\forall\ c_{in} \in \bullet n : Att_k(c_{in}) \neq \{\perp\}$. $\forall\ c_{out} \in n\bullet$:

1. if n is an activity node: $Att_{k+1}(c_{out}) = Att_k(c_{in}) \uplus Attributes(n)$
2. if n is an ANDjoin, ORjoin or XORjoin: $Att_{k+1}(c_{out}) = \uplus_{\forall c_{in} \in \bullet n} Att_k(c_{in})$
3. n is an ANDsplit, ORsplit or XORsplit: $Att_{k+1}(c_{out}) = Att_k(c_{in})$

The operator $\uplus$ represents the aggregation function applied when propagating the set of attributes. This operator will be discussed later in this paper.

The attributes propagation operates by propagating the set of attributes on the connectors. It starts by an initialization step that assigns the value $\{\}$ to the outgoing connector of the $InitialNode$ and the value $\perp$ to the other connectors. If a connector $c \in C$ is annotated by $\{\}$, then this means that the set of propagated attributes from the $InitialNode$ until this connector is empty. If a connector $c \in C$ is annotated by $\perp$, then this means that the set of propagated attributes are not yet defined for that connector.

Going back to our running example depicted in Fig. 1, the initialization step makes $Att_0(\#1) = \{\}$ and all the other connectors will be initialized to $\perp$.

After the initialization step, the propagation is done by firing one node at a time. Each node n might introduce some changes on the set of propagated attributes from its incoming connector(s) $\bullet n$ and propagates them on its outgoing connector(s) $n\bullet$. If the fired node n is an ActivityNode then we compute the $\uplus$ of the propagated attributes from $\bullet n$ with the attributes of the fired node n. If the fired node n is a join node then we compute the $\uplus$ of all the attributes from all the incoming connectors $\bullet n$ of the fired node. If the current node n is a split then there are no changes on the set of attributes from $\bullet n$ and they are propagated as they are on the the outgoing connectors $n\bullet$.

The operator $\uplus$ represents the aggregation function applied when propagating the set of attributes. This function depends on the control flow pattern being considered, the attribute type and its value. In the next we will discuss such operator for each case.

There exist in the literature several attempts to determine the aggregation function for computing QoS parameters of composed web services using control flow patterns [17, 18]. Major aggregation functions used in such contributions are summation, average, maximum, etc. where all the values an attribute has are considered in the computed value. But in our work, if a propagated attribute has more than one value, the propagation function should consider either all the values or only one of the alternatives.

To select the right values for the aggregation we defined a control mechanism based on categorization of the attributes (i.e., each attribute is tagged by a category). Each category helps determine the required aggregation function. If we need to determine the aggregation function applied on an attribute, we simply need to indicate its category. In the following, we present the set of categories that we take into:

- *Dominant*: the value of an attribute of this category cannot change. During the aggregation operation if only one of the alternative values is dominant, then its value is the only one to consider. If multiple alternatives are dominant, then the attribute value becomes an enumeration of all the dominant values.
- *Composed*: the value of an attribute of this category depends on a function. Its aggregation consists of updating this function.
- *Passive*: the value of an attribute of this category can be overridden by any other value if it has a superior category (i.e., Dominant or Composed).

It is important to note that there is a superiority order between these categories: *Dominant* $>$ *Composed* $>$ *Passive*. These categories help to determine the right aggregation function from this list:

- *Copy*: simply copies the attribute without applying any changes.
- *Override*: overrides the value of the attribute and considers only the superior category (*Dominant* $>$ *Composed* $>$ *Passive*).
- *Enumerate*: makes the attribute an EnumerationValue and lists the possible values.
- *Conditional*: transforms the attribute value into a ConditionalValue.
- *Composition*: applies on attributes where a formula is needed to compute its value. This function consists of determining the new function of the aggregated attribute.

If the Fired Node Is an *ActivityNode* Where the Input Connector Is Not Guarded by a Condition. In order to determine the right aggregation function applied to compute the propagation of attributes when firing an ActivityNode n, we refer to Table 2. Each column corresponds to the category of the attribute $at \in Attributes(n)$ (i.e., dominant, passive and composed). Each line corresponds to the category of the same attribute $at \in Att_k(\bullet n)$. Each cell defines the right aggregation function that is needed.

Recall, we proceed now to the second iteration where the node to be fired is the ActivityNode *Checking Content*. As the connector #1 is annotated by $\{\}$, that means $\forall at \in Attributes(CheckingContent) : at \notin Att_1(\#1)$. According to Table 2, the required aggregation function is copy (at). The result of this iteration is reflected on the connector #2 that is annotated by the attributes of the ActivityNode *Checking Content*.

Table 2. The Required Aggregation Function when Firing an ActivityNode

	$at \in Attributes(n)$ of category **Dominant**	$at \in Attributes(n)$ of category **Passive**	$at \in Attributes(n)$ of category **Composed**	$at \notin Attributes(n)$
$at \in Att_k(\bullet n)$ of category **Dominant**	Enumeration(at) of category Dominant	Override(at) of category Dominant	Override(at) of category Dominant	Copy(at) of category Dominant
$at \in Att_k(\bullet n)$ of category **Passive**	Override(at) of category Dominant	Enumeration(at) of category Passive	Override(at) of category Composed	Copy(at) of category Passive
$at \in Att_k(\bullet n)$ of category **Composed**	Override(at) of category Dominant	Override(at) of category Composed	Composition (at) of category Composed	Copy(at) of category Composed
$at \notin Att_k(\bullet n)$	Copy (at) of category Dominant	Copy(at) of category Passive	Copy (at) of category Composed	

If the Fired Node Is *Split* Node (i.e., ANDsplit, ORsplit or XORsplit). If the fired node n is an ANDsplit, an ORsplit or a XORsplit, the aggregation function is always a Copy(at). In other words, each attribute $at \in Att_k(\bullet n)$ is copied to all its outgoing connectors. More formally: $\forall c \in n\bullet : Att_k(c) = Att_k(\bullet n)$.

The third iteration of our algorithm consists of firing the first XORsplit. The operation here is a simple copy operation. Both connectors #3 and #4 are now annotated with a copy of the attributes from the connector #2.

If the Fired Node Is an *ActivityNode* Where the Input Connector Is Guarded by a Condition. Table 2 defines the aggregation functions when firing an ActivityNode where $Cond_{\bullet n} = Condition(\bullet n) = \perp$. In other words, the input connector of the node n is not guarded by a condition $Cond_{\bullet n}$. If this is not the case (i.e., the connector is guarded by a condition $Cond_{\bullet n}$), we composed the pre-mentioned aggregation function from Table 2 with a Conditional function, where the condition is $Cond_{\bullet n}$. If an attribute $at \in Attributes(n)$ and $at \notin Att_k(\bullet n)$ then the propagated attribute will have a ConditionalValue on its copied value (i.e., line 4 of Table 2).

During the fourth iteration, the fired node is the ActivityNode *Selectivity Processing*. This ActivityNode introduces the attribute *TypeOfCheck*. According to Table 2, the aggregation function should be Copy(TypeOfCheck) as it is the case in the second iteration. However, the connector #4 is guarded by the condition *ExamDecision = imp:PhysicalInspectionRequired* which imposes also the aggregation function Conditional that makes the attribute TypeOfCheck a ConditionalValue where the condition is ExamDecision = imp:PhysicalInspectionRequired and its value would be an enumeration of RedCheck and GreenCheck.

If the Fired Node Is a *Join* Node (i.e., ANDjoin, ORjoin or XORjoin). The aggregation function depends on the category of the attributes $Att_k(\bullet n)$.

- If exactly 1 attribute $at \in \bigcup_{c \in \bullet n} Att_k(c)$ is of category **Dominant** (or **Composed**)
  - This attribute value overrides all the other alternative values and the resulting attribute is of category **Dominant** (or **Composed**)
- If there are several attributes $at \in \bigcup_{c \in \bullet n} Att_k(c)$ of category **Dominant** (or **Composed**)
  - The propagated attribute value will be an enumeration of all the alternative values and the resulting attribute is of category **Dominant** (or **Composed**)

- If there is no attribute $at \in \bigcup_{c \in \bullet n} Att_k(c)$ is of category **Dominant** or **Composed** (Only **passive** attributes)
  - The propagated attribute value will be an enumeration of all the alternative values and the resulting attribute is of category **Passive**

After nine iterations of our algorithm on the running example, the node to be fired is the last XORjoin node. We apply an Override aggregation function of the attributes from #3 and #10 which is actually in this case a simple union operation (because they shared attributes have the same values).

Listing 1.2 represents the set of attributes of the composed capability of the entire process models of Fig. 1. Together with the ActionVerb *Examination of Cargo*, we can interpret this capability as follows: This capability allows to **examin a cargo**, where an **examination decision** determines if the cargo has to be checked; if a **physical inspection is required** then an **X-Ray scan** is permformed; if a **physical inspection is required**, and the **type of check is a Red Check** then a *detailed examination* is done.

Listing 1.2. Composed Capability of the model depicted in Fig. 1

```
:Phil_Cap_RunningExample a cap:Capability;
cap:hasActionVerb imp:ExaminationOfCargo;
impc:hasCargo :Phil_Cargo;
impc:hasExamDecision :Phil_ExamDecision;
impc:hasExamType :Phil_ExamType;
impc:hasTypeOfCheck :Phil_TypeOfCheck.

:Phil_ExamDecision a impc:ExamDecision, cap:EnumerationValue;
cap:hasElement :PhysicalInspectionRequired;
cap:hasElement impc:PhysicalInspectionNotRequired;
cap:hasElement impc:PriorityChannel.

:Phil_ExamType a impc:ExamDecision, cap:EnumerationValue;
cap:hasElement [ a cap:ConditionalValue;
                 cap:hasCondition impc:PhysicalInspectionRequired;
                 cap:hasCondition impc:RedCheck;
                 cap:hasValue impc:Detailed. ];
cap:hasElement [ a cap:ConditionalValue;
                 cap:hasCondition impc:PhysicalInspectionRequired;
                 cap:hasValue impc:X-Ray. ];

:Phil_TypeOfCheck a impc:TypeOfCheck, cap:ConditionalValue;
cap:hasCondition impc:PhysicalInspectionRequired;
cap:hasValue [cap:EnumerationValue;
        cap:hasElement impc:RedCheck;
        cap:hasElement impc:GreenCheck.].
```

Apart from this running example, we evaluated our approach on a set of process models from the customs clearance processes, namely import procedures. The test collection that we have considered in this work includes ten business processes. They describe guidance on the basic regulatory requirements that all importers must consider when they plan to import goods. The import customs clearance involves various steps from submission of import documents until the release of the imported goods. This evaluation was a simple validation of the results of our implementation over a small set of process models. As part of our future work, we plan to go for an empirical study over various application domains while exploring a large process model repository.

4 Related Work

Business process models are central artifacts in Process Aware Information Systems. These models are being managed and maintained by several stakeholders with various needs. Business Process Model Abstraction (BPMA for short) is one of the possible techniques that allows to have a quick view of essential elements of process models depending on the required level of detail. We find it then useful to compare our work towards interesting contributions in this field.

Business Process Abstraction can be technically implemented via two operations: elimination and aggregation [19]. Elimination omits unwanted model elements [2], however, aggregation makes a process model more coarse-grained [3–8]. In the best case, the aggregation operation allows to transform an entire process model into a single high-level activity which is the aim our our work in this paper.

Aggregation relies mainly on structural transformation [3–7]. By structural model transformation we mean detecting possible aggregation candidates based on structural patterns. In this context, [3–5] elaborate advanced structural patterns, therefore, [6] limits aggregation candidates to blocks having single input/output.

Meronymy (part-of) relations between activity labels is investigated in [8] in order to capture granularity relation between activities at several levels of abstraction. We currently, use a similar approach for detecting the action verb of the composed capability of the entire process model.

While [8] limits the aggregated activity into a simple label, [6] consider more elaborated model properties. By model properties, authors refer to Quality of Service properties which does not feature capability aspects or Input Output Precondition and Effect which results into a complex logical expression that needs extensive analysis for a proper interpretation of the Precondition and Effect.

5 Conclusion

We have presented in this paper one possible technique that allows to derive a high level process description from a detailed model. We assumed for our work that the input model is an annotated process model. At some point our assumption might be questionable. Actually, it is very hard to find process models semantically annotated with the required level of details. This is actually a common problem for any work that requires semantically annotated process models.

In addition, our approach demands that there exists an ontology used while describing the process. The research performed in the FP6 SUPER project (WP8) (http://www.ip-super.org/) shows that users prefer using standards instead of a home-developed ontology. Moreover, when there is a thousand or more concepts within an ontology, the users get lost within the ontology and therefore descriptions they deliver are of poor quality. To address this problem, we plan to use Natural Language Processing (NLP) for providing an automation support when annotating process models using our conceptual model. We envision to make the process annotation phase more user-friendly hiding any complexity to the user by analyzing textual descriptions of process models.

Another prospective improvement of our approach consists of the revision of how do we find the action verb of the entire process model. It is not always possible that an

entire process model could be abstracted into a single function. This assumes that all process models are created with respect to a given ontology of actions. It is common that each company creates its own ontology of actions when conceptualizing its own capabilities models. When these models are shared within other partners, they should consider the original ontology of actions and not only the one they have. In other words, several ontologies of actions can be proposed for the same domain which imposes defining a more advanced method for finding the right action verb of the entire model.

References

1. Dumas, M., van der Aalst, W.M.P., ter Hofstede, A.H.M.: Process-Aware Information Systems: Bridging People and Software Through Process Technology. Wiley (2005)
2. Eshuis, R., Vonk, J., Grefen, P.: Transactional process views. In: Meersman, R., et al. (eds.) OTM 2011, Part I. LNCS, vol. 7044, pp. 119–136. Springer, Heidelberg (2011)
3. Reichert, M., Kolb, J., Bobrik, R., Bauer, T.: Enabling personalized visualization of large business processes through parameterizable views. In: SAC. ACM (2012)
4. Polyvyanyy, A., Smirnov, S., Weske, M.: Reducing Complexity of Large EPCs. In: MobIS. LNI, vol. 141, GI (2008)
5. Smirnov, S., Reijers, H.A., Weske, M.: A Semantic Approach for Business Process Model Abstraction. In: Mouratidis, H., Rolland, C. (eds.) CAiSE 2011. LNCS, vol. 6741, pp. 497–511. Springer, Heidelberg (2011)
6. Vulcu, G., Bhiri, S., Derguech, W., Ibanez, M.J.: Semantically-enabled business process models discovery. Int. J. of Business Process Integration and Management 5 (2011)
7. Cardoso, J., Sheth, A.P., Miller, J.A., Arnold, J., Kochut, K.: Quality of service for workflows and web service processes. J. Web Sem. 1(3) (2004)
8. Smirnov, S., Dijkman, R., Mendling, J., Weske, M.: Meronymy-Based Aggregation of Activities in Business Process Models. In: Parsons, J., Saeki, M., Shoval, P., Woo, C., Wand, Y. (eds.) ER 2010. LNCS, vol. 6412, pp. 1–14. Springer, Heidelberg (2010)
9. Bhiri, S., Derguech, W., Zaremba, M.: Modelling capabilities as attribute-featured entities. In: Cordeiro, J., Krempels, K.-H. (eds.) WEBIST 2012. LNBIP, vol. 140, pp. 70–85. Springer, Heidelberg (2013)
10. Roman, D., de Bruijn, J., Mocan, A., Lausen, H., Domingue, J., Bussler, C., Fensel, D.: WWW: WSMO, WSML, and WSMX in a Nutshell. In: Mizoguchi, R., Shi, Z.-Z., Giunchiglia, F. (eds.) ASWC 2006. LNCS, vol. 4185, pp. 516–522. Springer, Heidelberg (2006)
11. Martin, D., Paolucci, M., Wagner, M.: Bringing Semantic Annotations to Web Services: OWL-S from the SAWSDL Perspective. In: Aberer, K., et al. (eds.) ISWC/ASWC 2007. LNCS, vol. 4825, pp. 340–352. Springer, Heidelberg (2007)
12. Kopecký, J., Vitvar, T., Bournez, C., Farrell, J.: SAWSDL: Semantic Annotations for WSDL and XML Schema. IEEE Internet Computing 11(6) (2007)
13. Lathem, J., Gomadam, K., Sheth, A.P.: SA-REST and (S)mashups: Adding Semantics to RESTful Services. In: ICSC. IEEE Computer Society (2007)
14. Oaks, P., ter Hofstede, A.H.M., Edmond, D.: Capabilities: Describing what services can do. In: Orlowska, M.E., Weerawarana, S., Papazoglou, M.P., Yang, J. (eds.) ICSOC 2003. LNCS, vol. 2910, pp. 1–16. Springer, Heidelberg (2003)
15. Malone, T.W., Crowston, K., Herman, G. (eds.): Organizing Business Knowledge: The MIT Process Handbook. MIT Press, Cambridge (2003)

16. Liu, R., Kumar, A.: An Analysis and Taxonomy of Unstructured Workflows. In: van der Aalst, W.M.P., Benatallah, B., Casati, F., Curbera, F. (eds.) BPM 2005. LNCS, vol. 3649, pp. 268–284. Springer, Heidelberg (2005)
17. Jaeger, M.C., Rojec-Goldmann, G., Mühl, G.: QoS Aggregation for Web Service Composition using Workflow Patterns. In: EDOC. IEEE Computer Society (2004)
18. Karaenke, P., Leukel, J.: Towards ontology-based qos aggregation for composite web services. In: GI Jahrestagung (1). LNI, vol. 175, GI (2010)
19. Smirnov, S., Reijers, H.A., Weske, M., Nugteren, T.: Business process model abstraction: a definition, catalog, and survey. Distributed and Parallel Databases 30(1) (2012)

Ontology-Based Big Dimension Modeling in Data Warehouse Schema Design

Xiufeng Liu[1] and Nadeem Iftikhar[2]

[1] Department of Computer Science, Aalborg University, Denmark
xiliu@cs.aau.dk
[2] Technology & Business, University College of Northern Denmark
naif@ucn.dk

Abstract. During data warehouse schema design, designers often encounter how to model *big dimensions* that typically contain a large number of attributes and records. To investigate effective approaches for modeling big dimensions is necessary in order to achieve better query performance, with respect to response time. In most cases, the big dimension modeling process is complicated since it usually requires accurate description of business semantics, multiple design revisions and comprehensive testings. In this paper, we present the design methods for modeling big dimensions, which include horizontal partitioning, vertical partitioning and their hybrid. We formalize the design methods, and propose an algorithm that describes the modeling process from an OWL ontology to a data warehouse schema. In addition, this paper also presents an effective ontology-based tool to automate the modeling process. The tool can automatically generate the data warehouse schema from the ontology of describing the terms and business semantics for the big dimension. In case of any change in the requirements, we only need to modify the ontology, and re-generate the schema using the tool. This paper also evaluates the proposed methods based on sample sales data mart.

Keywords: OWL ontology, Big dimension design, DW schema, Partitioning based design methods.

1 Introduction

Nowadays data warehouses (DW) are widely used for analysis and decision making. The data in DW is organized into dimension and fact tables. A fact table is typically linked to several dimension tables by foreign keys so that users can view their business data from different perspectives. With the increasing complexity of today's businesses, a growing number of dimensions and dimensional attributes are added into the multi-dimensional data model in order to answer complicate business questions. It becomes common to see a dimension table populated with hundreds of attributes. Furthermore, the size of data in the dimension table can also be very large (note that the size of a dimension table is typically much smaller than of a fact table). A typical example is the customer dimension in the sales data mart (Fig. 1), which possibly contains millions of customer records. The sales data mart is a well-known sample database schema, published by Inmon [15] in 1997. The customer dimension is commonly known as a big

W. Abramowicz (Ed.): BIS 2013, LNBIP 157, pp. 75–87, 2013.

dimension. The relational implementation of the multi-dimensional data model with big dimensions can lead to a high query response time since it possibly has to fetch a lot of unnecessary data when reading records. For this reason, a good schema design for big dimensions is critical important for query performance. In database technologies, data partitioning is widely used, and plays an important role to optimize database management systems. We believe that this technology is also applicable to the schema design of big dimensions. Thus, we can create the partitioned-based data model for big dimensions based on different partitioning techniques (Section 2). As a result, the data of big dimensions can be split and stored into multiple (partitioned) dimension tables. In this way, the queries retrieve data from the relative small dimension tables, which results in better query performance with minimum response time. In order to better illustrate the modeling process, we introduce the following running example in this paper.

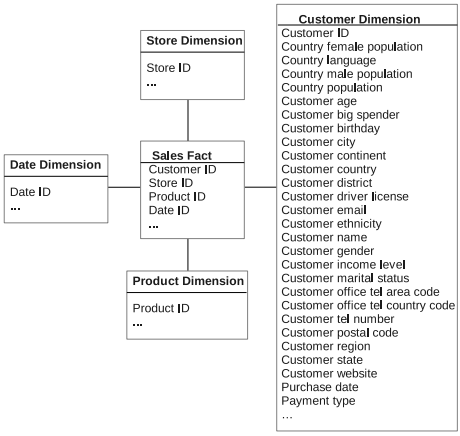


Fig. 1. The *sales* star schema with a big customer dimension table

Running Example. This example considers the star schema of the classical Inmon's sales data mart, which consists of a fact table (*Sales Fact*, see Fig. 1), and four dimension tables (*Date Dimension*, *Product Dimension*, *Store Dimension* and *Customer Dimension*, see Fig. 1). A fact table is the primary table in a dimensional data model where the numerical measurements of business are stored. A dimension table contains the textual descriptions of the business. The reason that we use this example, instead of the TPC-H [16], is that this famous Inmon's sales schema contains a big dimension, (*Customer*). It is a pure textbook star-schema (unlike TPC-H) and has been studied enormously. The Customer dimension consists of 53 attributes, and possibly holds millions of dimension records, which is unmanageable for querying purposes. To address this issue, the big dimension may be partitioned into multiple small dimensions based on columns and/or rows in order to improve the performance of most common queries. For instance, high-change-frequency attributes (Section 2.1) can be fragmented into a separate dimension. Similarly, the rows with a particular range (Section 2.1) can also be fragmented as a separate dimension. However, this manual modeling process is usually time consuming and tiresome. It raises the following challenges: *(i)* how to achieve a

good design for big dimensions. Since, we always notice that lot of modeling effort is spent on designing a fact table, whereas, too little attention is paid on designing dimension tables, due to the fact that their size (data volume) is much smaller than fact table. However, the design for big dimensions is also important, for the reason that it is a non-trivial task, which requires a good understanding of business, context and requirements. It also involves comprehensive testing to achieve a good design. As a result, the whole modeling process has to be repeated even with a slight change in the business requirements; *(ii)* how to automate the DW modeling process – in order to make it flexible, easy-to-use and efficient; and *(iii)* how to formalize the design methods – in order to standardize and to ensure the correctness of the DW modeling process.

Contributions. In this paper, we focus on all the above three mentioned challenges. We first use an OWL ontology to describe the semantics of a big dimension. The reason we employ OWL as the utility, instead of XML, UML or others, is that OWL supports the semantic reasoning, and is better for future extensions of this work, such as, reasoning-based DW schema design. In this paper, we formalize the big dimension design methods that include horizontal partitioning, vertical partitioning and the hybrid of both and propose the algorithm that describes the modeling process from an OWL ontology to a DW schema. We use the proposed design methods to achieve a good design for the big dimension (big dimension design examples in Section 2). We present a tool to automate the modeling process based on an ontology that describes the semantics of the big dimension. In this regard, our approach streamlines the modeling process from conceptual to physical DW design. Thus, we simply need to create/modify the ontology that describes the semantics of the big dimension and the tool creates/re-creates the data warehouse schema. The creation/modification of the ontology is not the focus of this paper for that reason it is not discussed.

The structure of this paper is as follows. Section 2 explains the design methods in detail. Section 3 describes the big dimension schema generation process using the proposed tool. Section 4 evaluates the proposed design methods. Section 5 discusses the related work. Finally, Section 6 concludes the paper and discusses the future works.

2 The Design Methods

In this section, we provide the details of the design methods and the modeling process for big dimensions. The modeling process includes choosing the design methods for big dimensions, describing the conceptual model of design methods using the OWL Lite ontologies and finally generating the relational database schema based on the model. We consider the following partitioning technologies as the baseline for big dimensions design: *vertical partitioning*; *horizontal partitioning* [8,3,2]; and their *hybrid* [12]. The vertical partitioning splits a big dimension table into multiple dimension tables vertically, each of which holds the same number of rows, but less columns. It is used to prevent a big dimension table from over-expanding, horizontally. The horizontal partitioning involves splitting the rows horizontally based on the values of one or more attributes. Each of the partitioned tables only holds a part of the rows. This method is used to prevent a big dimension table over-expanding, vertically. The hybrid partitioning combines the above two partitioning methods. This method results in more dimension tables, but each of them has less rows and columns.

2.1 Formalization

In this section, we formalize the design methods in order to standardize and to ensure the correctness of the DW modeling process. To be general, we presume that there is a big dimension in the business world, which is described by an ontology O. O is a collection of classes, data type properties and object properties described as below:

$$O = R_O\{C, DTP, OTP\} \tag{1}$$

where:

- C is a set of OWL classes;
- DTP is a set of data type properties;
- OTP is a set of object type properties;
- R_O represents the transformed relational model over C, DTP and OTP;

For any $dtp_i \in DTP$, there exists a domain $c_i = D(dtp_i)$ where $c_i \in C$, and a range $Rng(dtp_i) \in DT_{xml}$ where DT_{xml} is the collection of XML Schema data types. For any $otp_i \in OTP$, there exists domain $c_i = D(otp_i)$ and range $c_j = Rng(otp_i)$ where $c_i, c_j \in C$, and $i \neq j$. For any two classes with inheritance relationship, e.g., c_j inherits c_i, they are represented as $c_j = SC(c_i)$. To transform an input ontology to a relational DW schema, the condition of $C \neq \emptyset$ is required, otherwise, nothing will be done. The classes $c_i \in C$ are disjoint, meaning that each table in the DW schema is generated from an OWL class in C. The properties, DTP and OTP, can be either an empty or non-empty set. The properties in both sets are generated as attributes in DW tables. We formalize the generated relational schema S as a collection of tables and attributes:

$$S = R_R\{T, A\} \tag{2}$$

where:

- T is a set of generated dimension tables;
- A is a set of attributes;
- R_R is a relation over T and A;
- t_i is a table in T;
- The attribute set of table t_i is denoted as $A_i \sqsubseteq t_i$;
- a_i is an attribute in A_i, denoted as $a_i \in A_i$;
- V_i is the set of values of an attribute a_i;
- v_i is a value of a_i if and only if $v_i \in V_i$;

Here, we assume T and A both are non-empty sets. That is, there exists at least one table, and each of the tables contains at least one attribute. We now use Algorithm 1 to describe schema generation process from an OWL ontology to a DW schema, involving a big dimension. Algorithm 1 works as follows. First, for every class in the ontology, a table is created with the same name as an OWL class, and an attribute representing the primary key is added into this table (lines 1–4). Second, for all data type properties the algorithm finds the domain class and the range class, respectively. An attribute is created with the same name as the data type property. The attribute is added to the table which is mapped to the domain class, and the attribute data type is obtained from

the range of the data type property (lines 5–10). Third, for an object type property, the algorithm creates a foreign key relationship between the table that maps the domain class and the table that maps the range class. An attribute is, thus added to the table that maps the domain class and the foreign key constraint is added to this attribute as well (lines 11–18). Last, attributes are added to the tables that map OWL subClasses, through an inheritance statement referring to their parent table (lines 19–24).

Algorithm 1. Schema generation process from an OWL ontology to a DW schema of a big dimension

Require: Ontology of the form $O = R_O\{C, DTP, OTP\}$
Ensure: Relational Schema: $S = R_R\{T, A\}$
1: **for all** classes $c_i \in C$ **do**
2: Generate the table t_i from c_i, where $t_i \in T$
3: Create the primary key a_{PK} of t_i
4: **end for**
5: **for all** datatype properties $dtp_i \in DTP$ **do**
6: $c_j := D(dtp_i)$, $c_j \in C$
7: $\diamond$ Find the corresponding table $t_j \in T$
8: $rng := Rng(dtp_i)$, $rng \in DT_{xml}$
9: Create $a_i \in A_j$, where $A_j \sqsubseteq t_j$
10: **end for**
11: **for all** object properties $otp_i \in OTP$ **do**
12: $c_j := D(otp_i)$, $c_j \in C$
13: $\diamond$ Find the corresponding table $t_j \in T$
14: $c_i := Rng(otp_i)$, $c_i \in C$
15: $\diamond$ Find the corresponding table $t_i \in T$
16: Add an attribute a_{FK} to t_j
17: Add the foreign key constraint FK to a_{FK} (referencing to a_{PK} of t_i)
18: **end for**
19: **for all** classes $c_i \in C$, $c_i = SC(c_i)$ **do**
20: Find the corresponding tables t_i, $t_j \in T$
21: **for all** $a_j \in A_j$, $A_j \sqsubseteq t_j$ **do**
22: create $a_i \in A_i$, $a_i = a_j$ s.t. $A_i \sqsubseteq t_i$
23: **end for**
24: **end for**

Vertical Partitioning. Vertical partitioning is performed by splitting a big dimension table into multiple tables, each of which contains different number of columns. We conduct vertical partitioning by considering the following conditions: *(i) performance,* for the reason that when DW DBMS query a vertical partitioned table, less data is paged into main memory at a given time; and *(ii) change history,* since some values might change more frequently than others, the big dimension is split into multiple tables according to the changing frequency of attribute values. For example, we can classify the attributes of the Customer dimension in the running example into *high-frequent-changing* and *low-frequent-changing* attributes, respectively. Thus, updates and queries can proceed on a partitioned table with less columns.

We now formalize the vertical partitioning with a given OWL ontology O as follows:

$$O = R_{O1}\{c_1, dtp_1\} \cup R_{O2}\{c_2, dtp_2\} \cup ... \cup R_{On}\{c_n, dtp_n\} \tag{3}$$

where $D(dtp_i) = c_i$. This definition is derived from (1) given that C is the set of c_i, $i = \overline{1,n}$, and DTP is a non-empty set of dtp_i, $i = \overline{1,n}$. Given n disjoint OWL classes, O is transformed into the relational schema, S, described as below:

$$S = R_{R1}\{t_1, A_1\} \cup R_{R2}\{t_2, A_2\} \cup ... \cup R_{Rn}\{t_n, A_n\} \tag{4}$$

Equation (4) conforms to (2), where T is the set of t_i and A is the set of A_i, $i = \overline{1,n}$. Each of the OWL classes is thus mapped into a single partitioning table, whereas, each of the data type properties is mapped to an attribute of this table. To exemplify the partitioning, we split the Customer dimension table based on the changing frequency of the attribute values. The input ontology consists of the following two classes: *CustomerCouldChange* and *CustomerNeverChange*, which are eventually transformed into the two partitioned tables in DW DBMS (see Fig. 2).

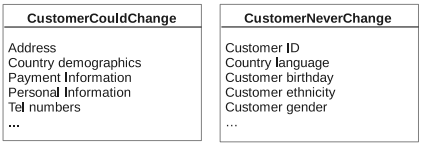


Fig. 2. Vertical partitioning

Horizontal Partitioning. In horizontal partitioning, a big dimension is split at least into two tables, each of which contains fewer rows but the same number of columns. The splitting is based on the values of one or more attributes. We partition the big dimension horizontally using the following approaches: *(i) range partitioning,* it partitions the rows according to the value intervals of an attribute or a set of attributes. For example, the record of a customer whose age satisfies *date* $>=$ *'2010-04-01' AND date* $<$ *'2010-05-01'* will be inserted into its corresponding partitioned table. The partitioned tables are disjointed; *(ii) list partitioning,* it is a partitioning technique where one can specify a list of discrete values for the partitioning key in the description for each partition. For example, a separate partitioned table is created for holding the information of the customers from a specific continent, for example, *continent = 'Europe'*; *(iii) hash partitioning,* a hash function is applied to the partitioning key values, and based on the hash values the ownership of a specific row is determined; and *(iv) round-robin partitioning,* in this approach the rows (to be inserted) are assigned to the partitioned tables in a round-robin fashion. This approach ensures that each partitioned table is to contain more or less equal number of rows. Based on the formula in (1), we formalize the horizontal partitioning as follows:

$$O = R_{O0}\{c, DTP\} \cup R_{O1}\{c_1\} \cup ... \cup R_{On}\{c_n\} \tag{5}$$

where $c_i = SC(c)$, $i = \overline{1,n}$. The OWL ontology defined by (5) contains an OWL class corresponding to the input dimension, and n subClasses derived from this OWL class.

The data type properties that are relevant to this OWL class are specified as well. The resulting relational schema is formalized as follows:

$$S = R_{R0}\{t, A\} \cup R_{R1}\{t_1, A\} \cup ... \cup R_{Rn}\{t_n, A\} \tag{6}$$

where a is an attribute from A, with the value v_j, $j = \overline{1,n}$. For each $R_{Ri}\{t_i, A\}$, a has the value v_i, $i = \overline{1,n}$. The n OWL subClasses are transformed into n partitioned tables in the relational database schema, each of which contains only one of the n possible values of attribute a. $R_{R0}\{t, A\}$ represents the table with all the attributes, while the other n partitioned tables will inherit all the attributes of this table. We call this table as *parent* or *master* table, while the other partitioned tables as *children*. This is done through the object relational feature of a DW DBMS, i.e., inheritance. Thus, each of the partitioned table only holds part of the rows, which are obtained from partitioning based on the attribute values of a, e.g, hash partitioning. The horizontal partitioning consists of the following steps: *(i)* it creates a master table, $R_{R0}\{t, A\}$, which all the partitioned tables will inherit; *(ii)* it creates all children, $R_{Ri}\{t_i, A\}$, $i = \overline{1,n}$; *(iii)* it adds table constraints to all the children, i.e., define the allowed key values for each child, v_i; *(iv)* it creates indexes in each table; and *(v)* it defines the trigger in the master table which is used to redirect the inserted rows to the corresponding child (note that all the insertions are done on the master table, instead on the children. Thus, the resulting partitioned schema is transparent to user). The aforementioned steps can be carried out automatically through the tool presented in this paper (we will discuss it in Section 3). The tool interprets the input OWL ontology. It extracts the necessary information for creating the tables, attributes and adds the inheritance relationship between the master table and its children. It also creates indexes and applies the *CHECK* constraints for row insertions. The tool-based transformation ensures the correctness of the big dimension modeling process. We now provide the details of the horizontal partitioning using the running example (Fig. 3). The Customer dimension table is partitioned based on the values of continent attribute. Thus, we can make the conceptual model of the horizontal partitioning for the Customer dimension using OWL constructs. The conceptual model contains six OWL classes, each of which corresponds to the customers from a specific continent. All the six OWL classes inherit a single *Customer* OWL class. In schema transformation, the proposed tool automatically transforms the six OWL classes into six dimension tables, i.e., *Customer_Continent_is_ Europe*, *Customer_Continent_is_Asia* and so forth. Since the six child tables inherit a single master table, the attributes of Customer dimension table are inherited as well (Fig. 3). When we perform the queries, we only need to query the master Customer dimension table directly, instead of the child tables. We use an object-relational database, such as PostgreSQL, as the DW DBMS, and create a *CHECK* constraint in the master table (see step (iii)). In this way, if we query the customer data from a specific content (given in *WHERE* clause), we only need to query the master table, whereas, the *CHECK* constraint (behind the scene) will automatically forward the query to the corresponding child table. Thus, the use of horizontal partitioning is transparent to end users.

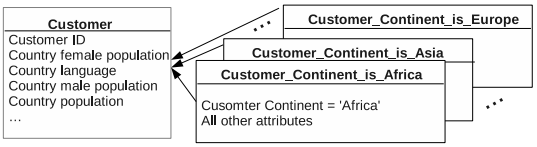


Fig. 3. Horizontal partitioning

Hybrid Partitioning. This method combines both the vertical and horizontal partitioning. We now formalize the ontology for hybrid partitioning:

$$\begin{aligned} O = & R'_{O0}\{c', dtp'\} \cup R'_{O1}\{c'_1\} \cup ... \cup R'_{On}\{c'_n\} \\ & \bigcup R''_{O0}\{c'', dtp''\} \cup R''_{O1}\{c''_1\} \cup ... \cup R''_{Om}\{c''_m\} \end{aligned} \tag{7}$$

For simplicity, equation (7) shows a union of two expressions described by (5). To be general, (7) can be expressed as the union of any number of equation (5).

$$\begin{aligned} S = & R'_{R0}\{t', A'\} \cup R'_{R1}\{t'_1, A'\} \cup ... \cup R'_{Rn}\{t'_n, A'\} \\ & \bigcup R''_{R0}\{t'', A''\} \cup R''_{R1}\{t''_1, A''\} \cup ... \cup R''_{Rm}\{t''_m, A''\} \end{aligned} \tag{8}$$

Equation (8) describes the resulting relational schema. For hybrid partitioning, we first partition the original big dimension into a number of vertical partitioning tables, then partition each of the tables horizontally into another set of tables, e.g., n and m tables shown in equation (8). The tables resulted from the vertical partitioning act as the master tables whose attributes are inherited by their child tables resulted from the horizontal partitioning. Fig. 4 shows the resulting relational schema for the Customer dimension using the hybrid partitioning. The vertical partitioning are based on the changing frequency of the attribute values of the Customer dimension table. Thus, it is partitioned into *CustomerNeverChange* and *CustomerCouldChange* tables, respectively. Based on the master table *CustomerNeverChange*, we perform horizontal partitioning on the values of the continent attribute. For the *CustomerCouldChange* master table, a number of child tables are created according to the age interval of customers, i.e., *CustomerCouldChange_Age_less_20*, *CustomerCouldChange_Age_between_21_40*, ..., *CustomerCouldChange_Age_more_81*. Similarly, the *CHECK* constraints are added into the master tables, so that querying the child partitioned tables should kept transparent to end users.

3 The Ontology-Based Tool

In this section, we present an ontology-based tool to automate the design process from conceptual to physical design for big dimensions (Fig. 5). To use this tool, users first need to create an ontology file that describes the conceptual model of a big dimension using OWL constructs. The creation of the ontology is not the focus of this paper for that reason it is not discussed. When the ontology is given as the input, the tool automatically generates the necessary SQL scripts, and further creates the schema in the underlying DW DBMS. The proposed tool supports three kinds of partitioning methods including

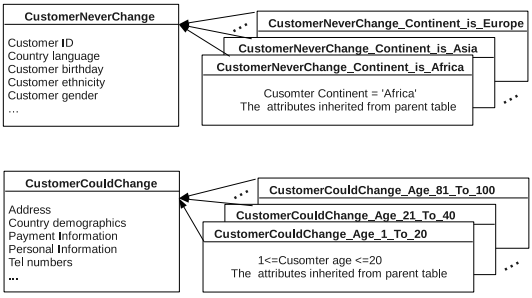


Fig. 4. Hybrid partitioning

horizontal, vertical and hybrid partitioning (Section 2). The tool retrieves the input ontology file that defines the conceptual model of the big dimension, and parses it using an ontology walker (*OWLOntologyWalker*). The walker visits all the ontology constructs, including OWL classes (*OWLClass*), subclass axioms (*OWLSubClassAxiom*), data type properties (*OWLDataProperty*) as well as object properties (*OWLObjectProperty*) and converts them into the corresponding constructs of the relational schema based on the mapping rules and naming conventions. The tool then automatically generates SQL scripts and executes them in the underlying DW DBMS. This makes it very convenient for the end users to do any revisions during the design, i.e., they only need to edit the input ontology and re-execute the program.

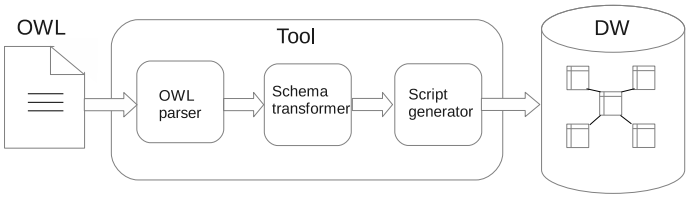


Fig. 5. The ontology-based big dimension schema generator

4 Evaluation

In this section, we measure the performance for each of the partitioning methods presented in this paper. We conduct the experiments on a Dell OptiPlex 960 workstation equipped with a 2.66 GHz Intel(R) Core(TM) 2 Quad processor, a 320 GB SATA hard drive (7200 rpm, 16 MB Cache and 3.0 Gb/s) and 3.0 GB RAM running with Fedora 14.0 Linux with kernel 2.6.35. The underlying DBMS PostgreSQL 8.3.5 uses the following settings: "shared_buffers=512 MB; temp_buffers= 128 MB; work_mem=56 MB, maintenance_work_mem=256 MB; checkpoint_segments=20"; and default values for other configuration parameters. We generate the synthetic data for the star schema of sales data mart (Fig. 1). The size of test data set for `Customer` dimension is 1,000,000 rows. We run eight queries (Q1–Q8, Table 1) to test the partitioning methods. Due to space limitations, we put the SQL scripts of the queries at `http://tinyurl.com/cng4b2y`.

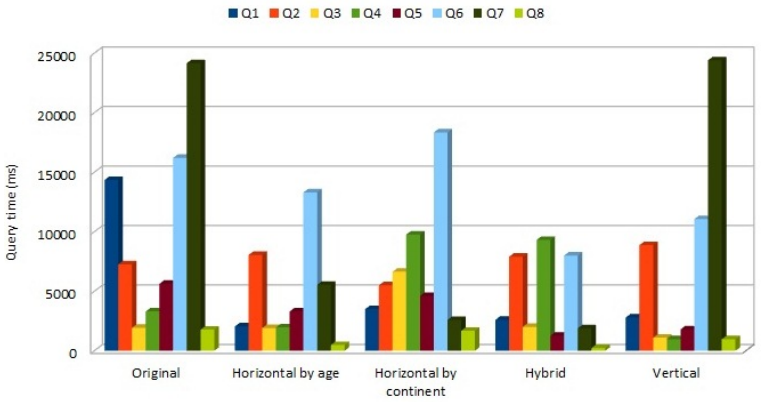


Fig. 6. Query performance for various partitioning methods

Table 1. The queries on the `Customer` dimension table with 1 million records

Query	Description	Affected rows
Q1	Select all the customers who are between 38 and 48 years old, and made their first purchase in the year $<$ 2000.	70,421
Q2	Select all the customers from Europe who are not Caucasians.	40,654
Q3	Select all the customers who speak French or Deutsch (as native speakers), live in France or Germany, cannot drive motorcycle, and bought a product with quantity $<$ 1000.	15,453
Q4	Select all the customers who have made at least one purchase in the last 20 years, and their purchase quantity $>=$ 1.	30,345
Q5	Select all the customers who are less than 18 years old, and their first names do not begin with a vowel.	38,542
Q6	Select all the customers who are between 38 and 48 years old, made their first purchases in the same year as the other customers whose age interval is “48-58”, and bought the same product in a calendar year between 2000 and 2005.	87,324
Q7	Select all the customers who are not from Europe, and made a purchase in the last 10 years with the quantity $>=$ 5.	186,546
Q8	Select all the customers who are less than 18 years old, and do not have driver license.	15,565

We run the same query for each partitioning method 10 times, and compute the average query response time in millisecond (ms). The results of each query are written to Linux null device, `/dev/null`. We run the eight queries for each of partitioning methods in order to evaluate the advantages and disadvantages of the proposed methods. In Fig. 6, we have a complete picture of all partitioning methods, as well as the behavior of the eight queries that were used. Q1 and Q2 highlight the advantages of the horizontal partitioning by age and continent. Q1 only searches one table (the one that corresponds to the group age 38-48). The worst behavior is obtained for the original schema (the one that is not partitioned). Similarly, Q2 searches the partition Europe only. The worst

time is obtained for the vertical schema, for the reason that Q2 must search through all customer entries just like in the case of the original schema, but does so with two tables (*CustomerNeverChange* and *CustomerCouldChange*). The time is worst due to the necessity of joining these two tables in order to obtain the result. Furthermore, Q3 and Q4 suggest that the vertical partitioning has the best behavior. Regarding Q3, all attributes that belong to the customer, are found in one table. The worst result is obtained for the horizontal partitioning by continent. Although, Q3 searches for customers that are from France or Germany, however, the search involves all tables, since the continent is not specified. Similarly with reference to Q4, the vertical partitioning again has the best result, as it only searches in the *CustomerCouldChange* table. In addition, Q5, Q6, Q7 and Q8 emphasis the significance of the hybrid partitioning. With regards to Q5, the original schema has the worst time, for the reason that Q5 performs the search on the entire table. Likewise, Q6 runs well on the hybrid partitioned by taking advantage of the age intervals (only require two tables to be searched) and fewer attributes due to the nature of vertical partitioning. With regards to Q6, the worst time is obtained by using the horizontal partitioning by continent. Moreover, Q7 runs slower on the vertical partitioning, since it needs to join all partitioned tables, whereas, the worst time for Q8 is obtained when running it on the original schema, as it searches through the entire set of rows. In conclusion, the hybrid partitioning has proved to be the best with respect to query time. Similarly, the horizontal and vertical partitionings have also performed well. The original schema for big dimension proved to be the worst.

5 Related Work

In the context of partitioning methodologies, there are many studies that propose different ways of partitioning. In general, partitioning techniques are part of the broad process of optimizing in a relational data warehouse [3]. The horizontal and vertical partitioning methods are different in the way the application perceives them [8]. Horizontal partitioning is a so-called transparent method with regards to the applications. While vertical partitioning affects the model and the way data is accessed. Vertical partitioning is also presented in [2], however, the work applies vertical partitioning to compare column-oriented DBMS with row-oriented DBMS. A mixed fragmentation based methodology in distributed databases is proposed by [12]. The methodology allows the optimal partitioning of global relations in a distributed database. The previous work presented so far provides the theoretical foundation for data partitioning. In this paper, we focus on big dimensions in a row-oriented DBMS execution environment (traditional DBMS, such as PostgreSQL) and propose an algorithm that describes the DW schema design process based on these partitioning methods.

Ontologies are used in many fields such as data integration, conceptual modeling as well as the semantic web. An approach to automatically discover meaningful IDs (composite keys) from domain ontologies is presented in [1]. The work presented in this paper is some what similar to this approach in a sense that we also use domain ontologies to automate the DW schema generation process. Work on ontology-based operational database schema generation have also been done [4,6,10,17]. An ontology-based (semi-automatic) approach for multidimensional DW design is presented in [14].

The approach searches for a specific multidimensional concept (as potential dimensions and facts) and requires user interaction for each step. In contrast to all these approaches, our approach is specific to the modeling of big dimensions in data warehouse schema design using different design methods. In addition, our approach reduces user interaction to minimum and with the help of the tool proposed in this paper, thus, the ontology-based schema transformation can proceed automatically.

Furthermore, some works focusing on big dimensions have been found. A data warehouse striping technique in distributed data warehouses is proposed in [7]. This data partitioning technique partitions the fact tables through all nodes and replicates the dimension tables. This technique is based on the idea of reducing the size of the fact tables as well as the size of the dimension tables. The technique to some extend decreases the number of the rows in the dimension tables, however, it does not focus on decreasing the number of columns. In contrast, our approach is based on a centralized data warehouse, and it is capable of reducing the dimensions based either on row or on column, as well as their hybrid.

6 Conclusion and Future Work

This paper proposed a flexible modeling approach for big dimensions using OWL ontologies. Although a number of OWL to DW schema transformation techniques have been proposed, however, to the best of our knowledge, this work is the first to present the design methods for big dimensions using horizontal, vertical and hybrid partitioning. The work formalizes the design methods and presents a tool to automate the modeling process. The proposed solution does not require any hand coding; instead it auto-generates the DW schema based on a given OWL ontology. The solution is easy-to-use and easy-to-maintain by end users. It is general and works well where the big dimension requirements are not fixed. Furthermore, some challenging areas for future work are as follows. Currently, the tool takes only one OWL ontology file as the input, and handles the schema generation with respect to the design of one big dimension. In future it should be extended to handle multiple big dimensions. In addition, a more complex improvement refers to a unified OWL schema as the input. The tool would automatically find the best method to partition the big dimension, or the desired partitioning method would be given as part of the input. Currently, the tool only has the reasoning capability on the instances of a class and its sub-classes. It would be desirable to integrate with a reasoner, which makes it to support more complex semantic requirements of DW schema design.

References

1. Abello, A., Romero, O.: Using Ontologies to Discover Facts IDs. In: DOLAP, pp. 3–10 (2010)
2. Abadi, D.J., Madden, S.R., Hachem, N.: Column-Store vs. Row-Stores: How Different Are They Really? In: SIGMOD, pp. 1–14 (2008)
3. Agrawal, S., Narasayya, V.R., Yang, B.: Integrating Vertical and Horizontal Partitioning into Automated Physical Database Design. In: SIGMOD, pp. 359–370 (2004)

4. Astrova, I., Korda, N., Kalja, A.: Storing OWL Ontologies in SQL Relational Databases. ECSE 1(4), 167–172 (2007)
5. Eberhart, A.: Automatic Generation of Java/SQL based Inference Engines from RDF Schema and RuleML. In: Horrocks, I., Hendler, J. (eds.) ISWC 2002. LNCS, vol. 2342, pp. 102–116. Springer, Heidelberg (2002)
6. Gali, A., Chen, C.X., Claypool, K.T., Uceda-Sosa, R.: From Ontology to Relational Databases. In: ER Workshops, pp. 278–289 (2004)
7. Costa, M., Madeira, H.: Handling Big Dimensions in Distributed Data Warehouses using the DWS Technique. In: DOLAP, pp. 31–37 (2004)
8. Imhoff, C., Galemmo, N., Geiger, J.G.: Mastering Data Warehouse Design: Relational and Dimensional Techniques, pp. 285–317. John Wiley and Sons, NY (2003)
9. Kalyanpur, A., Pastor, D.J., Battle, S., Padget, J.: Automatic Mapping of OWL Ontologies into Java. In: SEKE, pp. 98–103 (2004)
10. Liu, X., Thomsen, C., Pedersen, T.B.: 3XL: Supporting Efficient Operations on Very Large OWL Lite Triple-stores. Information Systems 36(4), 765–781 (2011)
11. Moody, D.L., Kortink, M.A.R.: From Er Models to Dimensional Models II: Advanced Design Issues. Business Intelligence Journal 8(4) (2003)
12. Navathe, S.: A Mixed Fragmentation Methodology For Initial Distributed Database Design. Journal of Computer and Software Engineering 3(4), 395–426 (1995)
13. Owl Description, `www.w3.org/TR/2004/REC-owl-features-20040210` (September 20, 2012)
14. Romero, O., Abello, A.: Automating Multidimensional Design from Ontologies. In: DOLAP, pp. 1–8 (2007)
15. Silverston, L., Inmon, W.H., Graziano, K.: The Data Model Resource Book: A Library of Logical Data Models and Data Warehouse Designs. John Wiley and Sons, NY (1997)
16. TPC-H, `http://tpc.org/tpch/` (September 20, 2012)
17. Vysniauskas, E., Nemuraite, L.: Transforming Ontology Representation from OWL to Relational Database. Information Technology and Control 35(3A), 333–343 (2006)

Utilizing Structured Information from Multiple External Sources in the Context of the Multidimensional Data Model

Matthias Mertens[1], Tobias Krahn[1], and H.-Jürgen Appelrath[2]

[1] OFFIS - Institute for Information Technology, 26121 Oldenburg, Germany
{mertens,krahn}@offis.de
[2] University of Oldenburg, 26129 Oldenburg, Germany
appelrath@informatik.uni-oldenburg.de

Abstract. Analytical Information Systems (AIS) enable analysts to visualize and analyze large amounts of data. They are based on a Data Warehouse in whose context typically various types of metadata are used. Often these metadata slightly consider additional information especially concerning the Multidimensional Data Model (MDM) like definitions, business rules, terminology or background information. By adding this metadata, particularly regarding the linked data movement, a significant improvement in the domain of AIS can be achieved. Our approach suggests a semantic metadata layer that enhances the AIS to allow modeling additional information in form of real-world entities. These entities correlate with MDM elements and are derived and integrated from various external structured sources. As a prototype we show the feasibility of this approach through a filter component that filters classification nodes with information not covered by the MDM.

Keywords: Analytical Information System, Semantic Metadata Layer, Data Warehouse, Linked Data, Multidimensional Data Model, Ontology.

1 Introduction

A key requirement of Business Intelligence (BI) is to improve the decision making process and to empower business users to get all the needed information at the right time. Among concepts, methods, and tools that focus on communication, collaboration as well as document and knowledge management, model and method based Analytical Information Systems (AIS) were developed as expert systems. They enable analysts to visualize and analyze large amounts of data sets. Based on a Data Warehouse (DWH), which integrates various information sources in a quality assured and multidimensional manner, analysis components allow to do Online Analytical Processing (OLAP) operations and complex statistical or geographic procedures in different visualizations like pivot tables, diagrams or thematic maps. However, AIS suffer from several shortcomings, especially in the context of an Information-Self-Service in particular for less skilled business users.

W. Abramowicz (Ed.): BIS 2013, LNBIP 157, pp. 88–99, 2013.

The First shortcoming results from the flexibility and mightiness of AIS. They are typically too complex for most business users. They face notable challenges performing adequate ad-hoc analysis, in a self-service manner, for receiving answers to their questions. In distinction from analysts, business users ordinarily neither have a deep technical understanding of the underlying Multidimensional Data Model (MDM), nor do they have the analysis knowledge in order to know which measures on which dimensions, aggregation levels and visualizations have to be analyzed in which analysis step to answer their analytical questions. Furthermore, the usage of the right OLAP operations and visualizations requires corresponding domain specific analysis knowledge[9].

Another shortcoming of AIS is the lack of additional supporting metadata associated with the integrated quantitative and qualitative DWH data, especially according to the DWH structure (measures or dimension elements), such as assumptions, definitions, business rules, terminology and background information [10]. Therefore, users have to exploit the semantics of the data and the structure on their own and often have to supply themselves with additional external data [5]. Furthermore, the use of external structured linked data for analysis support is not considered in AIS at the moment [3][9].

A third considerable problem, related to the second, results from the fact of having no possibility to import, capture, process, and use explicit analysis and domain knowledge of analysts for further analysis support of the business users. Of particular interest is empirical knowledge such as strategies or best practices of retrieving answers to specific issues in the context of analysis in a specific domain. In current systems, the applied knowledge gets lost.[9]

"Ideally, business users should be empowered to analyze multidimensional quantitative data structures without having all the necessary above mentioned knowledge and without being technically skilled. This could be achieved by reducing the systems' complexity: The system should be able to support business users in their ad-hoc analysis by using modeled semantic machine readable and reasonable knowledge provided by a semantic metadata layer for advanced analysis assistance. The functions for a business user Information-Self-Service can reach from advanced information to selected MDM entities over a navigation support on semi-defined analysis paths to a recommendation system that suggests possible further analysis steps" [9].

While shortcoming 1 and 3 are in the focus of our paper [9], we discuss in this paper a new elementary part of our ontology-based semantic metadata layer for an AIS that allows new functions for the Information-Self-Service of business users. The focus here lays on the above mentioned shortcoming concerning the lack of additional supporting metadata. Therefore, we explain how experts can model generic context elements that allow saving further information in the form of attributes and relations to and between real-world entities represented in the MDM of the underling DWH [2]. Based on that metadata layer we show how experts can integrate different, additional structured data sources like DBpedia [3] as representative for Semantic Web respectively linked data sources and the "German structured quality reports" as an example for relational database

sources. On Top of that instantiated metadata layer we present as example a filtering component that allows business users to filter classification nodes of the MDM by using the loaded additional informations.

The remainder of this paper is organized as follows. After this introduction, our conceptual solution in form of the semantic metadata layer is presented in section 2. Based on this, we explain our prototype and the results of our evaluation in section 3. Section 4 presents related work in this research area before section 5 summarizes our results and points out further work.

2 Concepts

Often Metadata is defined as additional information about content, structure, context, and semantics of data as well as process information for the use of the data [1]. Within the scope of data processing, Metadata describes all the information needed for analyzing, layouting, and constructing of information systems [8]. Especially the terminology Metadata includes technical terms, terminologies as well as domain specific knowledge and context information, for example business management measures such as designations, distinction, provenance or usage.

In past projects, we have observed a lack of additional supporting metadata associated with qualitative DWH data, especially the MDM structure (measures and dimension elements), like terminology and further background information. The MDM partly covers the real-world. But specific information is defined implicitly by the dimension taxonomy or by the data of the hypercubes like relations of classification nodes of different dimensions. Frequently additional information exists about modeled entities outside the MDM, e.g. in the original data source of the DWH that is not covered by the Extraction-Transformation-Load process. Furthermore, additional information from structured sources like DBpedia can be used for information visualization and in additional analysis tasks.

Our approach is to model generic context elements as an ontology that allows to save further information in the form of attributes and relations to and between real-world entities represented in the MDM and externally. Therefore, we need to model and instantiate the MDM in our semantic metadata layer in a way that it can be referred by the context-elements. This can be done as follows and shown in Fig. 1.

The section “Multidimensional Data Model Elements” and its eponymous entity is used to reference the elements modeled in the MDM. Therefore, the entity “Dimension” facilitates the deployment of a connection to an instantiated dimension, e.g. a “Hospital” dimension. A dimension consists of “Classification Levels”, which are composed of “Classification Nodes”. As an example, a classification level can aggregate hospitals as instantiated classification nodes in terms of the corresponding administration. Besides this, an MDM consists of further elements that are slightly indicated in Fig. 1 but not of importance for the forthcoming aspects.

After the instantiation of the named entities as an image of the MDM in our semantic metadata layer, we can refer to them by specific context elements

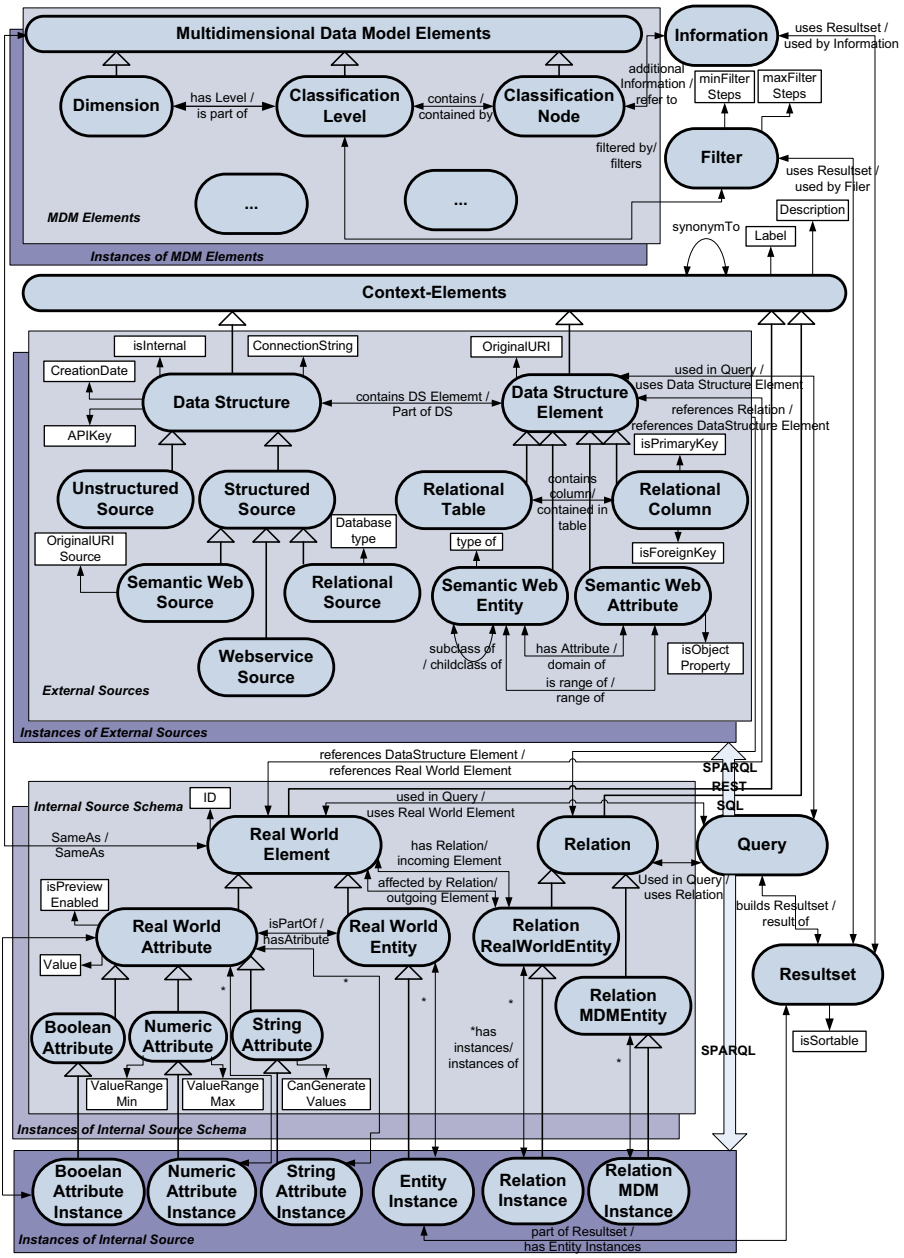


Fig. 1. Context Elements Ontology

describing additional information. The entity "Context-Elements" is the base entity for elements describing further information in the context of the MDM. All these elements inherit from this entity. As Fig. 1 shows, the "Context-Elements" in our ontology are divided into the following areas.

In the area "External Sources", entities, relations between them, and attributes are modeled on a high abstraction level. They allow the description of sources with different types. These sources contain additional information on the same real-world entities represented by the MDM through entities like measures, dimensions, classification levels and classification nodes. However, the real-world entities are not restricted to these MDM elements. For each source, the corresponding schema with relevant elements can be taken into consideration.

- The entity describing different sources outside the MDM is "Data Structure". These sources can be divided into structured and unstructured sources.
- Often, additional relevant information can be found in unstructured sources like wikis, blogs or other web based services, as well as in text based files located anywhere in the local network. To associate these unstructured data with the MDM or analysis based on the MDM is a challenge that is not in focus of this paper.
- In addition to unstructured sources, structured sources on relational, semantic web or web service basis exist describing the same real-world entities like the MDM does. Often they include further or alternative attributes, and also relations to entities outside the MDM that must also be considered in certain analysis tasks. As instances of these sources, the following examples can be mentioned: "DBpedia" as a representative of "Semantic Web Sources", "GeoNames" as a "Web Service Source", and the "German structured quality reports" as an instance of a "Relational Source".
- "Data Structure Elements" as part of the named "Data Structure" element are linked via the relation "contains DS Element" and inherited relations for each source type. This concept allows the modeling of specific source schema elements and the representation of instances that can be used in queries later on. The purpose is to know which instance of a source schema element and the corresponding source delivers specific information to attributes or relations of real-world entities. Hereby, "Data Structure Elements" are divided into "Relational Table" and "Relational Column", which are related to each other and to "Relational Source" as well. "Semantic Web Entity" and "Semantic Web Attribute" are connected to "Semantic Web Source".

The region "Internal Source Schema" defines a vocabulary that allows modeling real-world elements like entities, their relations to each other, and their attributes. Instances of "Internal Source Schema" are derived by experts based on the different external "Data Structure Elements" whereby one Instances can be composed of more then one source. This is especially the case, if each covers different aspects of the real-world element. These elements cover a part of the real-world that is partly included in the MDM and mostly go beyond. This is the reason why the entities "Real World Element" and "Relation" inherit from

"Context Elements" as they can model additional information which might be of interest.

- "Real World Element" is the parent entity of "Real World Entity" and "Real World Attribute". This entity is linked to "Data Structure Element" with the purpose that "Real World Element" instances like a "Hospital" or a "PostalCode Region" contained by defined instances of the external sources can be adopted. Typically, they are identified by a unique attribute "ID" like a "PostalCode" or a "Hospital ID".
- A "Real World Entity" allows the modeling of the mentioned instances whereby a real-world entity can be represented in more than one external source. Therefore, they are merged based on the unique identifier at instance level.
- Each "Real World Entity" has many "Real World Attribute" instances that can also be derived and adopted from external sources. Instances of this entity could be "area" linked with the instance "PostalCode Region" or "is University Hospital" and "medical devices" both linked with the instance "Hospital".
- Instances of "Real World Entity" can be linked to each other via instances of "Relation Real World Entity", which means that there is a specific relationship with explicit semantics between both. Hereby, the relation defines a domain and range as well as an inverse relation. Other attributes like cardinality, symmetry, functionality or transitivity are also possible. In some cases relations between instances of "Real World Attributes" are defined also to e.g. model dependencies. It should be noted that some sources, especially instances of "Semantic Web Source", already define named relations . Instances of "Relational Source" typically model relations between different entities by key relations.

In summary, instances modeled by vocabulary of the "Internal Source Schema" represent an excerpt of the real world that is also contained partly by the MDM. For example, the MDM models a "Hospital"- or a "PostalCode Region"-dimension. A real-world "Hospital"- and "PostalCode Region"-entity is contained by the relational source "German structured quality reports". The latter is also contained with additional informations in the "DBpedia" source, as well as in the "GeoNames" source. Instances of modeled instances of the "Internal Source Schema" like concrete hospitals or postal code regions, as well as others, their attributes and relations must be instantiated as an quality assured internal source. Often, these instances are equivalent to classification nodes, if they are also represented by the MDM but offer additional information for the later usage within an "Information"- and "Filter"-component. This bottom "Internal Source Schema" layer is build by experts who automatically tool supported query the different sources in their corresponding languages by utilizing the previously modeled metadata. The real-world instances are loaded and integrated in the bottom layer. Hereby typical integration aspects must be taken into consideration (see section 3). Afterwards they are matched and mapped against the instances of the MDM.

In addition, there are four other entities that are multi-connected to several entities and sections described above. These entities are not assignable to any section and use the other entities in a comprehensive way. By instantiating the "Filter" entity, the "Classification Nodes" of corresponding "Classification Level" can be filtered by specific "Real World Attributes" instances gathered from external sources. Furthermore, attributes defined by transitive linked real-world entities can be taken into consideration for filtering the classification nodes. If a "Filter" is instantiated as a logical filter, then typically this filter is characterized by executing more than one filter step whereby each step uses one instance of "Real World Attribute". A filter uses a "ResultSet", which consists of entity instances and is the outcome of a "Query". If more than one filter step is applied, the result set depends on the selected logic operation, like e.g. "union set" or "intersection set" that extends or reduces the classification nodes. With the aid of instances of the entity "Real World Element", instances of "Relation" and the modeled connection to instances of "Data Structure Element", it is possible to set up a query against the bottom "Internal Source Schema" layer.
Our ontology includes an "Information" entity that uses the result set and provides additional information for classification nodes. By instantiating this entity, contextual information can be made available to the user of the AIS like extra information on hospitals, e.g. number of beds or specialization of a hospital. The aim is to compensate the less available technical understanding and domain knowledge of business user.

After presenting the metadata model on a high abstraction level the next section 3 explains more practically the instantiation of the semantic metadata layer by experts and the resulting information-self-service for business users.

3 Prototype and Evaluation

With the "Multidimensional Statistical Data Analysis Engine" (MUSTANG) [12], the OFFIS Institute for Information Technology has created a framework for the development of AIS. MUSTANG is designed on a service-oriented architecture to enable explorative and ad-hoc multidimensional data analysis with the focus on geographical and statistical aspects provided by corresponding services. Today, MUSTANG is the foundation for several analytical applications in the German health care system [12], whereby the addressed shortcomings could be observed. Therefore, MUSTANG was enhanced in prototypical implementation with the described metadata model of section 2 within a semantic metadata layer [9], in particular the instantiation and filtering services.

First of all, experts instantiate the MDM-elements section of our ontology by using diverse MUSTANG core services. The goal is to correlate the MDM with additional information. In the next step, the External Sources" section as well as the "Internal Source Schema" section are instantiated in parallel. Therefore, an expert defines the utilizing sources and contained real-world elements iteratively. Hereby several aspect concerning the availability, origin, privacy, reliability as well as quality like granularity, topicality, correctness and completeness of data

should be considered. So, a source that is build by these experts on their own or in the company environment could be higher rated regarding these aspects. Often internal and external sources can distinguished whereby semantic web sources belong to the second group and are rated lower regarding some aspects.

For relational sources the user specifies entities and appoints the corresponding relational tables that describe with there columns literal attributes. Based on keys these columns can also describe relations to other real-world entities. Instances of Child Classes of "Data-Structure Element" and "Real World Element" are created. The experts provide statements if these "Real World Element"-Instances correlate with MDM-Elements on Instance Level so that a matching should be carried out. For example, he specifies the "German structured quality reports" source that contains additional information according to the real-world entities "Hospitals" correlating to a "Hospital" dimension of the MDM.

For semantic web sources like "DBpedia" also "Data-Structure Element"- and "Real World Element"-Instances are created, whereby a new problem may arise. If the user is not familiar with the schema, the corresponding data model and its capabilities it could be difficult to find the right "Data Structure Elements" for instantiation. In case of "DBpedia" there are 359 hierarchically arranged classes with 1775 Attributes[1] that could be inherited. In contrast to relational sources we propose a instantiation by example. Therefore a user appoints a concrete instance of the class which should be modeled as "semantic Web Entity"-Instance and the corresponding "Real World Entity"-Instance. All available and filled literal attributes and relations to other entities are suggested and the user could accept or reject the modeling as "Real World Element"-Instances. As another characteristic there could be synonymous sources like for example "dbpedia.org" and "de.dbpedia.org". Most entities are identical but there are differences in the vocabulary, covered attributes and relations and completeness of data. Therefore, attributes can be marked as synonymous. In particular, English language attributes such as "administrative district", point to other entities and are more valuable compared to their synonymous German equivalent literal values like in this case "Landkreis". Homonyms are not supported.

While modeling "Data-Structure Element" and "Real World Element"-Instances supported by our instantiation tool, the expert users should only model additions for still existing "Real World Element"-Instances to avoid heterogeneities. However if they model synonym attributes and relations defined by different sources, they have to specify the more trustworthy. The other synonym attributes and relations could be used for correctness and completeness checks.

With the adoption of the real-world entities from different sources as instances of the bottom layer "Instances of Internal Source Schema", the last step is processed. The instantiation tool processes the sources "Real World Element"-Instances in the modeled order and queries the corresponding "Data Structure Element"-Instance to extract the "Entity"-Instances. Synonym "Entity"-Instances from different sources are identified by a unique key that was modeled before. A combination of matching techniques that are length-, distance- or tokenbased as well as

[1] `http://wiki.dbpedia.org/Ontology`, as of 21.01.2013.

QGram and Jaro-Winkler could be applied. These techniques are also used to identify the equivalent MDM-Instances and for equivalent textbased literal attribute values. The process is supervised by the user who decides in case of doubt.

After the building of our semantic metadata layer the integrated informations can be used by the business users in form our prototypical filter component shown in Fig. 2. Its purpose is to filter classification nodes of a selected Dimension on a chosen classification level based on the mapped "Entity Instances". Therefore the business user has to process as workflow the columns from the far left to the far right column whereby iterations are possible.

In the first column on the far left, all classification nodes of a chosen classification level are displayed. In addition to this, the user can see how many classification nodes the chosen classification level contains and how the corresponding dimension is called. In this case, the selected classification level contains 11569 municipality of Germany in a "Region"-Dimension.

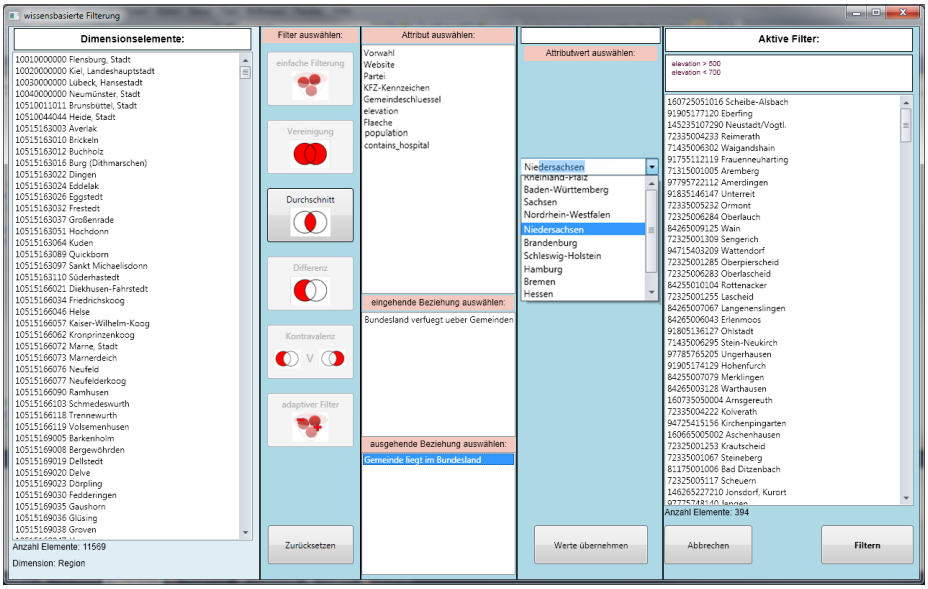


Fig. 2. Filter Component

To the right in the next column, an available filter type can be chosen. As the screenshot shows, in this scenario the business user selected the "intersection"-filter as a logical filtering operation, which means that each of the following filter attributes have to hit a classification node.

In the next column at the center of the user interface, all available filter attributes according to the actual chosen classification level are shown. In the example, all metadata of each filter attribute is gathered from different data

sources, describing one real-world entity: If the user chooses "contains hospital", the extracted informations of the "German structured quality reports" are used. In the case that he selects "elevation", integrated data from "DBpedia" is used. If the user selects "Gemeinde liegt in Bundesland", what means "federal state of the municipality", as a relation to the real-world entity "federal state" , data from "GeoNames" is used to filter the classification nodes. In case of relations also attributes of the referenced real-world entity could be used as transitive filtering instead of selecting an instance (not shown in Fig.2).

In the fourth column, a value according to the chosen filter attribute has to be selected. Because the user selected "federal state of the municipality" as a filter attribute, all available German federal states from "GeoNames" are displayed for selection. For transitive filtering the third column is reused with a new Tab. The user has to go one column to the left again.

In the last column on the far right, the current filter results are shown. The user selected the intersection filter and already filtered the classification nodes with the filter attribute "elevation" between 500 and 700 meters in a previous filter step. Depending on which federal state the user will chose in the current step (fourth column), the intersection set will be recalculated. By clicking the "Filtern"-Button the filtering process ends and the classification nodes of the resultset are adopted as valid Dice OLAP-Operation in the AIS. Alternatively more filter-steps beginning in column three can be applied.

In a evaluation of this prototype we gave six business users the task to manually select classification nodes of a "Hospital"-Dimension in the AIS that represent concrete hospitals with specific attributes. The underlying MDM contains about 1600 German hospitals that are hierarchically ordered by the ownership. The asked attributes where care of Cardiosurgical Centers and as region Lower Saxony. In the first run the test persons had to use the ordinary AIS and in the second run the enhanced prototype AIS. In the first run the test persons searched in individually chosen sources like "google" or "weisse liste portal"[2] and then selected the corresponding classification nodes. In second run they selected the information by using the filtering component. We discovered that our filtering component was better regarding the correctness and quickness of filtering results. As disadvantage we must mention the completeness and quality of context informations. Especially in case of semantic web sources and their open world assumption the filter result may not include all values.

Finally, we were able to summarize in a concluding expert conversation with a person from the epidemiological cancer registry in Lower Saxony that such a filter component for supporting business users is of high practical relevance because frequently projects require additional information that was not considered when the MDM was developed. In their scenarios especially potential hazards in municipalities and postal-code regions like power poles, power plants or landfills are of interest.

[2] A Portal in the German health care market based on the "German structured quality reports".

4 Related Approaches

There are some works in the literature that deal with similar research topics. As we have shown in this paper, we can integrate arbitrary structured sources like DBpedia or GeoNames, e.g. to filter classification nodes. Data from DBpedia is also used in [4] to extend the keyword search for business entities in the DWH by including metadata like synonyms or other commonly used expressions matching a given keyword. In [6] the MDM is extended by domain and mathematical ontologies to describe underlying mathematical elements of data cubes, and in [13] the authors built a consistent business semantics model by representing business metadata with an extended OWL language that is used to query the DWH. The authors of [5] use an ontology-based approach to bring different analysis roles together, especially business users, focusing on the collaboration between the participants of the analysis process. Another work of the "Self Service Business Intelligence" [7] initiative is [11] with the aim to improve the self-service capabilities by an ontology-based architecture and frontend-tool to ease accessing and querying the data. Overall, we can summarize that there are several works in the literature integrating BI- and Semantic Web-technologies - but barely any approach extends the MDM by adding structured information from multiple external sources.

5 Conclusion

In this paper, we have shown that the utilization of structured information from multiple external sources in the context of the Multidimensional Data Model has a great potential for the analysis support for users of an Analytical Information System. Due to the linked data movement in the last years, new capabilities for advanced data usage and analysis arise. In this context, two different efforts could be identified in the literature. On the one hand linked data sources like DBpedia are used to enhance Business Intelligence Systems. On the other hand, by using Semantic Web technologies within Analytical Information Systems, new analysis features come up. However, the combination of these two research areas is rarely taken into consideration. From our point of view, this approach offers huge potential as well as new challenges to be examined. On the basis of our described context ontology as a part of the semantic metadata layer, our approach offers possibilities to integrate relevant information concerning real-world entities from several external structured sources in a quality assured manner. These information are used for filtering classification nodes within the Multidimensional Data Model as well as for extended information supply to ease the analytical process. As an alternative to the creation of an internal source, "on the fly" queries can be created and sent to external data sources with the possible disadvantage of lower quality and lower performance. The implementation and evaluation of our approach in the form of a prototype and concluding expert conversation has proven successful concerning the usage of additional information for advanced analysis support. The belonging master thesis was honored with the first prize of the German TDWI Award 2012. As further work we suggest the combination of this type of metadata with pursuing business semantics such as business rules.

References

1. Auth, G.: Prozessorientierte Organisation des Metadatenmanagements für Data-Warehouse-Systeme. Books on Demand, Norderstedt (2004)
2. Bauer, A., Günzel, H.: Data Warehouse Systeme – Architektur, Entwicklung, Anwendung, 3. Auflage edition. dpunkt, Heidelberg (2009)
3. Bizer, C., Heath, T., Berners-Lee, T.: Linked Data: Principles and State of the Art. In: World Wide Web Internet and Web Information Systems (April 2008)
4. Blunschi, L., Jossen, C., Kossmann, D., Mori, M., Stockinger, K.: Data-Thirsty Business Analysts need SODA – Search Over Data Warehouse. In: Proceedings of the 20th ACM International Conference on Information and Knowledge Management, CIKM 2011, pp. 2525–2528. ACM, New York (2011)
5. Berthold, H., Rösch, P., Zöller, S., Wortmann, F., Carenini, A., Campbell, S., Bisson, P., Strohmaier, F.: An Architecture for Ad-hoc and Collaborative Business Intelligence. In: Proceedings of the 2010 EDBT/ICDT Workshops, EDBT 2010, pp. 13:1–13:6. ACM, New York (2010)
6. Diamantini, C., Potena, D.: Semantic Enrichment of Strategic Datacubes. In: DOLAP 2008: Proceeding of the ACM 11th International Workshop on Data Warehousing and OLAP, pp. 81–88. ACM, New York (2008)
7. Imhoff, C., White, C.: Self-Service Business Intelligence – Empowering Users to Generate Insights. Tdwi best practices report, TDWI (2011)
8. Kemper, H.-G., Baars, H., Mehanna, W.: Business Intelligence – Grundlagen und praktische Anwendungen, 3. Auflage edition. Vieweg+Teubner, Wiesbaden (2010)
9. Mertens, M., Krahn, T.: Knowledge Based Business Intelligence for Business User Information Self-Service. In: Brüggemann, S., d'Amato, C. (eds.) Collaboration and the Semantic Web, pp. 271–296. IGI Global, Hershey (2012)
10. ONeil, B.: Semantics and Business. The Data Administration (2007)
11. Spahn, M., Kleb, J., Grimm, S., Scheidl, S.: Supporting business intelligence by providing ontology-based end-user information self-service. In: OBI, p. 10 (2008)
12. Teiken, Y., Rohde, M., Mertens, M.: MUSTANG: Realisierung eines Analytischen Informationssystems im Kontext der Gesundheitsberichtserstattung. In: Informatik 2010: Service Science – Neue Perspektiven für die Informatik. CEUR Workshop Proceedings, pp. 53–68 (2010)
13. Xie, G.T., Yang, Y., Liu, S., Qiu, Z., Pan, Y., Zhou, X.: EIAW: Towards a Business-Friendly Data Warehouse Using Semantic Web Technologies. In: Aberer, K., et al. (eds.) ISWC/ASWC 2007. LNCS, vol. 4825, pp. 857–870. Springer, Heidelberg (2007)

Understanding the Impact of E-Commerce Software on the Adoption of Structured Data on the Web

Kurt Uwe Stoll, Mouzhi Ge, and Martin Hepp

Universität der Bundeswehr München
E-Business & Web Science Research Group
Werner-Heisenberg-Weg 39, 85577 Neubiberg, Germany
{uwe.stoll,mouzhi.ge}@unibw.de, mhepp@computer.org

Abstract. In this paper, we analyze the potential impact of e-commerce software packages on the diffusion of markup for structured data on the Web. We argue that such an analysis must focus on the *product detail pages,* i.e. the "deep links" to individual items, rather than the pure number of shop sites, for assessing the potential. Based on (1) a systematic analysis of the popularity of 56 software packages for e-commerce sites among the Alexa list of the one million most popular Web sites, we (2) estimate the number of product detail pages for the respective shops and then (3) project the potential lever of each e-commerce package for the adoption of structured markup, assumed that adding support for structured data can be made a readily available feature of a respective software. Our results indicate that by adding a structured markup component to as little as seven popular e-commerce systems, we could instantly deploy structured data markup on nearly 90 % of the product detail pages among the one million most popular Web sites.

Keywords: E-Commerce, Semantic Web, RDFa, Microdata, Microformats, SEO, schema.org, GoodRelations, Technology Adoption.

1 Introduction

In the past few years, embedding structured data with e-commerce information into HTML content using RDFa[1], Microdata[2], or Microformats[3]has become a mainstream technique for Web shops. With GoodRelations[1], an established ontology is available that supports modeling of a wide range of e-commerce scenarios with structured data.It has recently been integrated into the schema.org standard [2]advocated by four major search engines.

For shop owners, publishing structured data is mainly motivated by the consumption of search engines like Google, which use it to enhance search results. From a search engine perspective, shop pages that contain structured data are preferable, as the extraction of important information like product name or priceis computationally expensive and error-prone.When pages include structured data markup, this task becomes less difficult and potentially more reliable.

[1] http://www.w3.org/TR/xhtml-rdfa-primer
[2] http://www.w3.org/TR/microdata/
[3] http://microformats.org/about

W. Abramowicz (Ed.): BIS 2013, LNBIP 157, pp. 100–112, 2013.

In the context of the paper, we define e-commerce software packages as software artifacts that allow merchants to operate an e-commerce site that presents products or services and supports a purchasing transaction, e.g. via shopping cart functionality. Many shop operators use *standard* e-commerce applications to run their sites, such as Magento[4], ATG[5]or Prestashop[6]. State-of-the-art solutions typically offer an extension mechanism that enablesadding third-party code to the resulting system. This allows developing extension modules for structured data, which then significantly reduce the cost and effort of adding structured data markup, because instead of editing the HTML template, the administrator will just have to download and install the respective module. In the past years, we have developed or helped others to develop several such modules, which have shown significant uptake. As of January 2013, there are more than 18,000 aggregated downloads for our modules for different e-commerce applications that add structured markup. Still, these 18,000 downloads represent only a small part of the total number of shop sites on the Web.

For advancing the field of Semantic Web technology in e-commerce, however, the adoption of structured data markup by a limited number of Web shops is not enough, as the resulting market coverage will not be sufficient in terms of the range of products, or the coverage of dealers withdifferent value propositions. A broad range of products is especially relevant for applications that aim at providing a consolidated view on a respective market segment. For instance, a product search engine based on structured data for digital cameras will only be useful if a substantial share of the market is represented.

Our core research question is how e-commerce packages and the respective market structures provide an effective lever for accelerating the diffusion of structured data markup so that Semantic Web applications become possible. In essence, we want to know the number of e-commerce sites that run standardized software packages, and the distribution properties of the number of products per shop site. This allows projecting the impact of adding structured data functionality to a comparatively small number of codebases of e-commerce packages. To our knowledge, no previous systematic analysis of these questionsexists, except for non-scientific studies from an industry perspective.

In this paper, we provide (1) a systematic analysis of the popularity of 56 e-commerce applications among the Alexa list of the one million most popular Web sites [3], (2) estimate the number of product detail pages for the respective shops and then (3) project the potential lever of each package for the adoption of structured data markup, assumed that adding support for structured data can be made a built-in feature of a respective application. Our results show that by adding a structured markup component to as little as *seven* popular Web shop applications could instantly add structured data to nearly 90 % of the product detail pages among the one million most popular Web sites. Our main contribution is estimating the impact of adding structured data functionality to a limited number of e-commerce packagesfor the amount

[4] http://www.magentocommerce.com
[5] http://www.oracle.com/us/products/applications/commerce/atg/
[6] http://www.prestashop.com/

of respective markup at Web scale, which resides mainly in the "deep link" part of Web sites.

The remainder of this paper is structured as follows: In section 2, we summarize related work. In section 3, we describe our methodology and provide details about our data collection and implementation. In section 4, we analyze the data and summarize the findings. In section 5, we evaluate the performance of the critical component for the detection of e-commerce systems, discuss potential limitations of our approach, and sketch future work. In section 6, we conclude and summarize our results.

2 Related Work

In this section, we summarize work related to our approach, which can be grouped into three categories:

Market Studies: Due to the very dynamic nature of the field, it is unfortunately inevitable to refer to non-scientific resources for some figures. For instance, Raju estimates that in 2012, there were 90.500 shops in the US earning more than $12,000 [4].

Since 2011, Robertshawhas been conducting a semi-annual analysis of the market shares of e-commerce systems[5]. According to his results, the eleven biggest e-commerce systems account for more than 80% of all sites. The service http://builtwith.comprovides ongoing reports of the popularity for a wide range of Web technologies, including e-commerce packages[6]. Unfortunately, the site only delivers relative market share data of the ten most popular e-commerce packages with respect to the top 1 Million sites sample, which is of limited value for our research. Note that all these reports do not take into account the size of the "deep link" part of shop sites but merely count the sites directly.

Functional Comparison: Beside the market studies, there are many analyst publications targeting e-commerce systems, mostly aimed at corporate audiences. Those publications put a stronger focus on the comparison of features and a strategic outlook on the regarded systems than e.g. on the number of deployments. For instance, in 2011Gartner [7] provideda report that maps different e-commerce systems into four clusters. Since the criteria for inclusion in the report are such that they exclude lower end solutions, which may account for a substantial amount of Web shops in the long tail, the study does not match our focus. In 2012, Forrester [8] provided a similar report ranking different solutions in terms of offering and strategic position, also excluding lower end solutions.

Structured Data for E-Commerce: The GoodRelations Web vocabulary [1]was the first broadly adopted Web ontology for exposing structured data with e-commerce information following the Semantic Web vision[9][10]. The adoption of GoodRelations was supported by a wealth of freely available tools for publishing and consuming respective data; for an overview, see [11].While so farthere is no comprehensive quantitative analysis of the amount and granularity of GoodRelations data in the wild, Ashraf et al. have provided a preliminary study on GoodReations usage patterns on the Web [12].Recently, GoodRelations has been officially integrated into schema.org [2]. Schema.org is a collaboration of Google, Yahoo!, Bing and Yandex to promote a set of stable structured data vocabularies. We estimate that as of this writing, at least

15,000 shop sites with a total of at least 20 million product detail pages include GoodRelations data.

To the best of our knowledge, there is no previous work that analyzes the impact of e-commerce systems on the availability of structured data.

3 Methodology, Approach, and Implementation

The basic rationale for our research is the following: We know from our previous development of extension modules that it is possible to modify respective shop software packages to automatically add the publication of structured data based on GoodRelations, and that in a manner that (1) requires only minimal configuration effort for the site owner and (2)is tolerant with regard to modifications of thestylesheets, themes, and HTML templates, or the installation of other modules. Our next goal is to add respective functionality to the core codebase of popular e-commerce packages so that the adoption of structured data does no longer depend on the manual installation of such extensions. If we succeed with that, a large number of shop sites will automatically add GoodRelations markup once they are updated to the next version of the system. This promises to be a huge lever for the implementation of the Semantic Web vision for e-commerce.Given that we have limited resources for implementing the idea, we need to know (1) how many e-commerce systems we should target and (2) which coverage we can achieve on the level of product detail pages. We expect the number of those pages to be Pareto-distributed, likewise than, e.g. firm sizes. Thus, covering a minor share of systems with structured data may have a high impact on the overall coverage of the market.

There are four layers of interest in this context, as Fig. 1 illustrates.

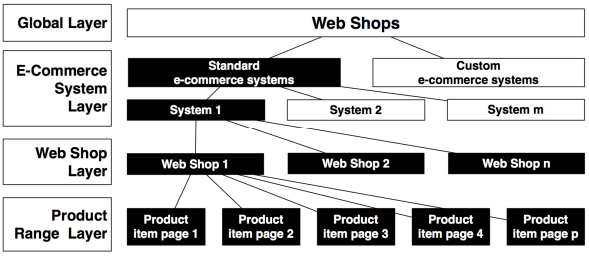


Fig. 1. Effect of enabling structured data for an e-commerce system on product pages

1. The global layer, which represents all shops on the Web.
2. The e-commerce system layer, which is divided into Web shops running standard e-commerce packages, and custom e-commerce systems based on proprietary software. In our work, we focus on standardized e-commerce systems, such as Magento, ATG, or Prestashop.
3. The third layer is the Web shop layer, constituted by the actual shop sites that run a specific e-commerce package or proprietary implementation.
4. The fourth layer is the product range layer. It consists of all the product detail pages hosted by a particular shop site and system.

3.1 Methodology

Our research approach consists of the following steps:

1. **Obtaining a list of relevant site URIs:** Since we cannot analyze the Web as a whole, we need a subset of URIs representing Web site main pages to start with. Roughly speaking, this is a list of Web sites, but not limited to e-commerce sites. We will screen them for e-commerce functionality in the subsequent steps. For our analysis, we take the freely provided Alexa Top 1 million traffic rank [18]. This gives us the URIs of the main pages of the one million most popular sites.
2. **Defining the shop software packages to search for:** As a second step,we need a list of relevant e-commerce software applications. For that purpose, we merged the top 40 list provided by Robertshaw [16], the top ten list from http://www.builtwith.comand the systems mentioned in the reportsby Gartner [2]and Forrester [3], resulting in a list of 56 search strings for e-commerce systems. This list is shown in Table 1.
3. **Determining whether a URI represents a Web shop using one of the systems from our list:** To get a hold of URIs that are run by specific e-commerce systems, we used the tool Whatweb [19], originally a site profile scanner from the context of computer security. Whatweb is able to detect a wide range of properties of a Web-site, including e-commerce functionality. We then matched the results against the list of search strings.
4. **Counting product item pages based on sitemaps:** Next, we need to estimate the number of product detail pages for each shop, which is a non-trivial challenge. As an approximation, we used XML sitemaps [20]of the shop sites, if available. In this context, we assumed the remaining sites to be a sufficient sample of the base population. We then conducted a cluster analysis on the sitemap properties to find the ratio of product item pages on a Web shop. We could show that product item pages and overall sitemap pages correlate. Thus, we use the URI counts based on sitemap in combination with the average share of product detail pages within a site as a first approximation of the number of products per shop.
5. **Extrapolation of the product item count to Web scale:**In order to predict the impact of equipping e-commerce systems with structured data markup, we project our results on the total number of shops in the population.
6. **Evaluation of the e-commerce system detection:** As the e-commerce system detection provided by Whatweb is a critical part of our analysis, we additionally evaluate its performance on a sample of n=550 URIs with human computation.

Table 1. Consolidatedlistofsearchstringsforthe56 e-commerepackages in the study

magento, zen cart, virtuemart, oscommerce, prestashop, opencart, volusion, Yahoo!stores, interspire, ubercart, wp e-commerce, ecshop, actinic, miva merchant, shopify, cs-cart, ibmwebsphere commerce, x-cart, oxidesales, 3dcart, atg, demandware, ejunkie, intershop, shopp, ablecommerce, nopcommerce, prostores, shopsite, foxycart, big cartel, ekmpowershop, gsi commerce, shopfactory, cubecart, roman-cart, tomatocart, drupal commerce, blucommerce, lemonstand, thefind upfront, google trusted store, cleverbridge, elastic path, icongo, jagged peak, marketlive, microsoft commerce server, netsuite, istore, venda, micros-retail, redprairie, digital river, sap e-commerce, xt-commerce

3.2 Implementation

Obtaining a list of relevant site URIs: The initial input to our study is the top one million list of Alexa[13]. Alexa analyzes website popularity. There is a monthly global ranking of top-level domains according to traffic (the "Top1m list"), provided for free in a CSV format. We used the 09/2012 release.

To understand whiche-commerce packages are used for the Web shops in the Top1m list, we employed the tool Whatweb[14]. Whatweb is an open-source security scanner written in Ruby. Among other site characteristics, it detects server software, content management systems, and e-commerce systems.

Unfortunately, applying a tool like Whatwebon such a large amount of URIs is computationally expensive. Thus, we employed cloud computing resources. While a common pattern is to distribute the task on many cloud instances, we found that running it parallelized on a single powerful machine was sufficient and resulted in the smallest overhead. We used the Amazon EC2 Cluster Compute Eight Extra Large cloud computing instance (cc2.8xlarge)[7]. We distributed four threads on each of the 16 cores of the machine using GNU parallel[8]with one line of code, which can be found in listing 1. Running the task took 8 hours and 32 minutes, resulting in server cost of 19.20 $ for the given 1 million URIs.

```
cat 1m.csv | parallels --max-threads=64 ruby whatWeb.rb> 1m.txt
```

Listing 1. Parallelization with GNU parallel

To get the subset of results related to the e-commerce packages of interest, we merged the top 40 list provided by Robertshaw, the top ten of builtwith.comand the leading systemsfrom the Gartner and Forresterreports, as already mentioned. The merged list of 56 search strings is given in Table 1 above.

For consistency, in the tables and figures of the remainder of the paper, we use the original lower case spellings of the search strings. We additionally considered the system *XT commerce*, as it was missing in the other surveys and is claimed to have more than 100,000 installations. We ran the list against the Whatweb results using a small script, matching the search strings against the Whatweb result file. It is important to stress that, to a certain degree, this approach is also able to detect shop systems even if there was no specific Whatweb plugin beforehand, as there are often strings hinting to shop systems in the part of the results of Whatweb (eg. cookies or HTTP headers).Those were not targetedby the original server detection plugins.

Counting product item pages based on XML sitemaps: After fetching and parsing the sitemaps, we went on to get a hold of product item pages. This figure is important, as web shops usually provide, beside the product detail pages(1) category pages, (2) review pages and (3) pages about payment and shipping options, to name a few. In order to assess the number of product detail pagesin a given Web shopwe assumed that the product item pages count should be correlated to the total URI count of the XML sitemap. To validate this, we conducted a k-means cluster analysis on the

[7] http://aws.amazon.com/en/ec2/#instance
[8] http://www.gnu.org/software/parallel/

properties of each entry of the sitemap of a sample of 716[9] randomly selected shops using Scikit-learn 2011[15]. We set the cluster size to three, as we assumed there would be a cluster of product pages, category pages, and of arbitrary pages. In preprocessing, we filtered URIs linking to images, and generated a property that yielded the existence of the string "product" in the server path. For the further analysis, we only took product, priority and lastmod properties into account. The resulting clusters were filtered to have a silhouette coefficient[16] of at least 0.6, and the relative size of the biggest cluster to be between 0.6 and 0.9 of the number of entries in the sitemap, as we considered only those who match this threshold as valuable sitemaps matching our initial assumptions.We then computed Pearson's correlation between the biggest cluster and the total sitemap page count. This resulted in a value of 0.879, indicating a strong correlation.Additionally, we computed a final correction factor that represents the mean difference between URIs found in a sitemap and its biggest cluster. The result is 0.774, with a 95% confidence interval of 0.759 to 0.790. Thus, in 95 % of the caes, there will be between 759 and 790 product item pages per 1000 URIs in a XML sitemap.

4 Results

4.1 Summary

Number of products per site: According to the correlation analysis conducted in section 3.2, we can take the URIs listed in a XML sitemap as an estimate for the number of product detail pages. The analysis of the XML sitemapsgives preliminary hints that the market for e-commerce systemsfollows a Pareto distribution with regard to the number of product detail pages, i.e. at the level of deep links. Six systems leading the URI count represent more than 90% of all URIs. The respective results are shown in table 2. Overall, 23.33 million URIs could be extracted from the XML sitemaps. If we apply the correction factor of 0.774 (see section 3.2), this projects to roughly 18 million product item pages.

Table 2. URIs found in sitemaps and product item estimate

	Shop software	**URIs**	Lowerboundaryof 95% confidenceinterval	**Projected # ofproduct item pages (n * 0.774)**	Upperboundaryof 95% confidenceinterval	**% ofproductsof all products**	**Cumulated % ofproducts**
1	Magento	12,610,254	9,571,183	9,760,336	9,962,101	54.05	54.05
2	ATG	3,016,552	2,289,563	2,334,811	2,383,076	12.93	66.98
3	Prestashop	2,756,334	2,092,058	2,133,402	2,177,504	11.81	78.80
4	osCommerce	1,597,558	1,212,547	1,236,509	1,262,071	6.85	85.64
5	Zen Cart	769,947	584,390	595,938	608,258	3.30	88.94
6	CS-Cart	524,778	398,307	406,178	414,575	2.25	**91.19**
7	Virtuemart	508,310	385,807	393,431	401,565	2.18	93.37
8	Others	1,546,366	1,173,692	1,196,887	1,221,629	6.63	100
Total		**23,330,099**	**17,707,545**	**18,057,496**	**18,430,778**		

Additionally, we visualized the findings using box plots[17], as shownin Fig. 2. For a higher expressiveness of the plot, we filter the systems to have product page counts in the 0.5 area of the standard deviation of each system's distribution, i.e. we

[9] This number emerged from n=50 samples per shopmaximum, if available.

filtered out extremely large (and small) sites. The resulting set of eight systems is the result of applying a filter so that only shops that have more than 50 results are considered. The boxes show the 50% quantile of the distributions after applying the filter above, the line in the box the median. The lines above and below the boxes are the whiskers, they show the remaining upper and lower 25% quantiles. Outliers are plotted as crosses. We can see that Demandware aggregates a high amount of URIs and additionally hashigh top 25% whiskers, whereas Virtuemart or Zen Cartdo not. As the median of the distributions is mostly located considerably towards the bottom of the 50% box, those systems spot a positive skew towards a low number of products. It also matches our informal experiences with maintaining various shop extensions.

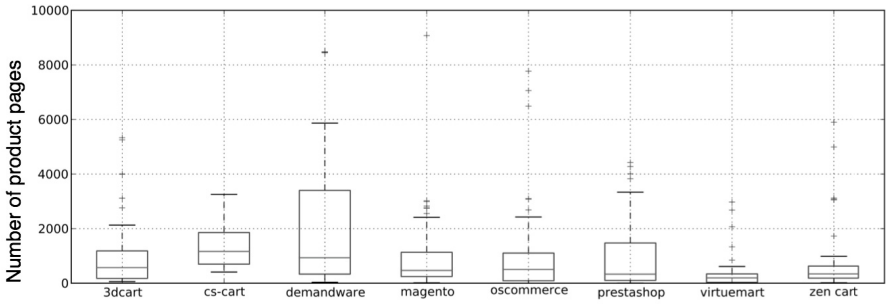


Fig. 2. Distribution of the number of product pages per shop software package

Market structures and popularity: The initial Whatwebexperiment resulted in 912,865 successful responses. Overall, 21,848 shops could be detected in the sample. The frequency count of the different e-commerce systems is shown in table 3. Only sixe-commerce systems cover more than 80% of regarded sample, andMagento leads the results with 34.7%.

Table 3. Popularityof e-commercesoftwareapplications

System	Magento	osCommerce	Prestashop	Virtuemart	Zen Cart	XT commerce	Opencart	CS-Cart	Others	SumURIs
URIs	7,582	2,764	2,629	2,144	2,129	1,131	545	455	2469	21,848
%	34.70	12.65	12.03	9.81	9.74	5.18	2.49	2.08	11.30	
sum %	34.70	47.35	59.39	69.20	78.95	**84.12**	86.62	86.62	88.70	

4.2 Impact of E-Commerce Software on the Adoption of Structured Data

Based on the tentative findings regarding the number of product detail pages and market structure, we can assume that adding structured data to the core codebases of only six e-commerce systems would already augment nearly 90 % of the product detail pages found in the sample with structured data markup. The systems with the highest impact would be Magento, osCommerce, ATG, Zen Cart, Prestashop, EC-Shop and Virtuemart. Except for ATG and EC-Shop,for all of these there are already GoodRelations extension modules available[18]. However, only a small share of shops actually use those extensions. Therefore, it should be a priority to add structured data functionality to the core codebase of shop systems. Another important activity will be the development of missing extensions.

4.3 Additional Findings

Site popularity: We analyzed the popularity of sites generated by specific e-commerce systems in terms of the Alexa traffic ranking. Herein, it is of interest which shop systems tend to be more present in the high-traffic sites, and which not. To answer this question, we chose to use the mean of each shops ranking distribution (AX-mean). To make the result more transparent, we provide an additional variable AX-factor, defined by dividing 500,000, i.e. the middle rank of the Alexa traffic ranking, with the mean of each shop. A higher value means higher ranking sites in average. Most shops ranked less than 1, which means that most of them position in the lower ranks of Alexa Top1m sites. Only ATG and Demandware yielded significant values of 2,7227 and 1,634, indicating that they are used by highly popular shops. A possible explanation is that really large shop applications use either proprietary code or employ technology components like load balancers that render the underlying e-commerce system hard to detect.

Sitemap availability and quality: Another observation is that many sites yielded incorrect sitemaps or provided none at all. To gain insight into this, we tried to fetch the sitemap according to robots.txt or the standard "sitemap.xml" path[19]. Then, we analyzed the results and counted occurrences of URIs. We additionally implemented an algorithm to parse sitemap indices. Only the previously low ranked 3DCart yielded significant positive results. The other shops achieved rates lower than 67%, down to, most often50%. This means the sitemap standard is notused properly in many cases, and results in high crawling effort for search engines. We expected high-end systems such as ATG to provide correct sitemaps, but their results were not better than those of the base sample.

Geographical distribution: Finally, we analyzed the geographical distribution of leading systems according to the number of products they manage. The United States, Germany, United Kingdom and France dominate the geographical distribution. The top ten countries of the seven most popular e-commerce systems are shown in Table 4.

Table 4. Geographical distribution of the systems according to the number of products

	Magento		ATG		Prestashop		osCommerce		ZenCart		CS-Cart		Virtuemart	
1	USA	**3062**	USA	**456**	FRAU	**950**	USA	**946**	USA	**1180**	USA	**226**	USA	**615**
2	GER	**974**	GER	**266**	USA	**436**	GER	**427**	UK	**116**	UK	**47**	GER	**305**
3	UK	**693**	UK	**44**	GER	**265**	FRA	**268**	NL	**110**	GER	**34**	RUS	**219**
4	FRA	**643**	FRA	**23**	SPA	**147**	UK	**239**	GER	**77**	AUS	**20**	FRA	**146**
5	NL	**371**	RUS	**19**	UK	**88**	POL	**99**	EST	**57**	NL	**15**	UK	**107**
6	BRA	**158**	CAN	**18**	POL	**68**	SPA	**90**	ROC	**49**	VN	**14**	NL	**65**
7	EU	**155**	TUR	**18**	EU	**52**	NL	**86**	EU	**38**	SA	**11**	UKR	**49**
8	AUS	**138**	NL	**17**	CZ	**48**	EU	**43**	MAL	**32**	EU	**9**	HU	**47**
9	SPA	**119**	EU	**16**	NL	**48**	AUS	**41**	JAP	**29**	POR	**8**	IT	**45**
10	IRE	**77**	POL	**12**	CAN	**42**	RUS	**41**	CZ	**28**	GR	**5**	POL	**45**

5 Evaluation of Shop Software Recognition

The e-commerce system detection is a critical part of our approach. We decided to assess the performance of the method using human computation via the service

Crowdflower[10], which is an intermediary providing access to a manifold of human computation services through a standardized interface. We use precision to evaluate the performance of information retrieval systems[20],as we cannot measure recall, because our approach is limited by the aforementioned list of systems. We set up a task for humans to decide whether a given URI is ane-commerce site or not. Thus, the experiment provides insight whether the list of shop URIs actually contains shop sites.

We ran the experiment for 11e-commerce systemsand presented to the human participants a list of 50 randomly selected URIs, resulting in 550 items to judge. According to the evaluation, the shop detection approach achieved a mean precision of 94%, i.e. the shops detected by Whatweb are actually shops. The systems we analyzed yielded a precision between 90% (e.g. ATG, Virtuemart) and 100% (e.g. 3DCart and Demandware). We show the results in Table 5.

Table 5. Reliability of the shop detection technique

Shop Software	3DCart	Demandware	Shopsite	Magento	Prestashop	CS-Cart	
Precision	100.00 %	100.00 %	97.00 %	96.00 %	93.00 %	92.00 %	
Shop Software	EC-Shop	osCommerce	Zen Cart	ATG	Virtuemart		**Mean**
Precision	92.00 %	92.00 %	92.00 %	90.00 %	90.00 %		94.00 %

6 Discussion and Conclusion

6.1 Limitations

Our work is subject to the following limitations:

1. Alexa Top1m as basis for the data collection induces is a bias towards popular sites. As future work, we plan to run Whatweb against the data of CommonCrawl[21], a public crawl of a substantial part of the Web. This would lower the bias towards popular sites and better represent the longtail of the Web.
2. We used Whatweb as it is, without additions to the plugins or constraining functionality. Improving the plugins could have resulted in higher performance in the site recognition process, but the overall result of our research is not dependent on marginal performance improvements of the underlying data collection. Constraining functionality of Whatweb in terms of excluding detection features would have resulted in a lower computational effort, but we would have lost additional data, which can be explored in future work.
3. E-commerce software missing in our initial links, additional components like load balancing tools, or weaknesses in the recall of our detection technique may account for a significant number of sites incorrectly excluded from our analysis. We evaluated this by drawing a sample of 100 sites of the Alexa list and manually judged whether they were shop sites. This resulted in a share of 21.74% e-commerce sites as compared to only 2.39% sites found by our technique. This may reflect a fundamental limitation of our quantitative results, unless the shop sites properly detected are a sufficiently representative sample of the overall situation. At this point, we do not know this.

[10] `http://www.crowdflower.com`

4. Another shortcoming might be the reliability of the string search over the results of Whatweb in order to detect the different shop systems.
5. The approach of using XML sitemaps to estimate the number of deep product detail pages is a limited technique. Many sites to not provide XML sitemaps and XML sites provided may list only a subset of actually available product item URIs. Alternative approaches for counting the number of product detail pages would be (1) deep crawling or (2) counting pages indexed by Google[11].In order to evaluate the quality of our approach, we conducted a deep crawl and requested the pages indexed by Google of n=13 randomly selected sites generated by five different shop packages. We then computed the correlation coefficients between the sitemap count and the two new heuristics. The results are shown in Table 6.

Table 6. Correlation of differerent page count heuristics

	Sitemap count / Deep crawl	Sitemap count / Google	Deep crawl / Google
Pearson's correlation	0.38	0.72	0.45

We see that neither (1) a sitemap count vs. a deep crawl nor(2) a deep crawl vs. Google index size correlate significantly. Only sitemap count vs. Google index size shows a more significant correlation of 0.72. To clarify those results, we manually inspected the URIs in the deep crawl. However, a deep crawl count without reliably detecting product detail pages seems to be a very questionable heuristic, as shops often (a) generate pages for every permutation of a category filter, and (b) provide pages for every review submitted by customers. This leads to giantURI lists from deep crawlsstemming from duplicate content.Also, the number of pages indexed by Google is a problematic estimate, as it depends on the index size and crawling budget Google allocates to a site.We further assessed the objection that the page count might differ between (1) sites that provide sitemaps and (2) sites that do not. An additional deep crawl on n=10 sites resulted in means of (1) 1,620.69 and (2) 2,110.20 and standard deviations of (1) 1,854.39 and (2) 4,543.26. This finding does not harm our initial result. To summarize, we think the approach taken is a fair technique given the limitations of alternative solutions.

6.2 Conclusion

In this paper, we presented an extensive analysis of Web shops within the one million most popular sites, in order to assess the impact of standardized e-commerce systemson the adoption of structured data markup for the Semantic Web. We can show that based on the high number of product detail pages, i.e. the "deep links" in shopping sites, adding structured data functionality to only six e-commerce software packages could add structured data markup to more than 90 % of all products detail pages in the sample. We have shown that working on the integration of the Semantic Web vision into those six software packages will likely be a very effective lever for the diffusion of structured data markup into real applications and to increase the market coverage in the resulting data.

[11] We requested the figure with Google site search, i.e. +site:`example.org`

Acknowledgments: The work on this paper has been supported by the German Federal Ministry of Research (BMBF) by a grant under the KMU Innovativ program as part of the Intelligent Match project (FKZ 01IS10022B), and by the Eurostars program (within the EU 7th Framework Program) of the European Commission in the context of the Ontology-based Product Data Management (OPDM) project (FKZ 01QE1113D).

References

1. Hepp, M.: GoodRelations: An Ontology for Describing Products and Services Offers on the Web. In: Gangemi, A., Euzenat, J. (eds.) EKAW 2008. LNCS (LNAI), vol. 5268, pp. 329–346. Springer, Heidelberg (2008)
2. http://schema.org/ (last checked on October 01, 2013)
3. Alexa Top 1 Million Sites By Traffic Rank As CSV (last checked on October 01, 2013)
4. How Many Online Stores Are There In The U.S.?, http://blog.referralcandy.com/2012/08/14/how-many-online-stores-are-there-in-the-u-s/ (last checked on October 01, 2013)
5. Robershaw, T.: October 2012 Ecommerce Survey, http://tomrobertshaw.net/2012/11/october-2012-ecommerce-survey/ (last checked on October 01, 2013)
6. Ecommerce Technology Web Usage Statistics, http://trends.builtwith.com/shop (last checked on October 01, 2013)
7. Alvarez, G., Fletcher, C., Sengar, P., Martz, S.A.: Magic Quadrant For E-Commerce. Gartner, Inc. (2011)
8. Walker, B.K.: The Forrester Wave (™): B2C Commerce Suites, Q3 2012. Forrester Research (2012)
9. Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web. Scientific American 284(5), 34–43 (2001)
10. Lassila, O.: Web Metadata: A Matter of Semantics. IEEE Internet Computing 2(4), 30–37 (1998)
11. Hepp, M., Radinger, A., Wechselberger, A., Stolz, A., Bingel, D., Irmscher, T., Mattern, M., Ostheim, T.: GoodRelations Tools and Applications. Poster and Demo Proceedings of the 8th International Semantic Web Conference (ISWC 2009), Washington, DC, USA (2009)
12. Ashraf, J., Cyganiak, R., O'riain, S., Hadzic, M.: Open Ebusiness Ontology Usage: Investigating Community Implementation of GoodRelations. In: Proceedings of the WWW2011 Workshop on Linked Data on the Web (LDOW 2011), vol. 813 (2011)
13. Can I Get A List Of Top Sites Using Web Services?, http://www.alexa.com/faqs/?p=35 (last checked on October 01, 2013)
14. Whatweb, http://www.morningstarsecurity.com/research/whatweb (last checked on October 01, 2013)
15. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., VanderPlas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-Learn: Machine Learning In Python. Journal of Machine Learning Research 12, 2825–2830 (2011)
16. Rousseeuw, P.: Silhouettes: A Graphical Aid To The Interpretation and Validation of Cluster Analysis. J. Comput. Appl. Math. 20(1), 53–65 (1987)
17. McGill, R., Tukey, J.W., Larsen, W.A.: Variations of Box Plots. The American Statistician 32(1), 12–16 (1978)

18. Adding GoodRelations To Standard Shop Software, http://wiki.goodrelations-vocabulary.org/Shop_extensions (last checked on October 01, 2013)
19. Sitemaps.Org - Protocol, http://www.sitemaps.org/protocol.html (last checked on October 01, 2013)
20. Manning, C.D., Raghavan, P., Schütze, H.: Introduction To Information Retrieval. Cambridge University Press (2008)
21. CommonCrawl, http://commoncrawl.org/ (last checked on October 01, 2013)

The Conception of the Model

Bernhard Thalheim

Christian-Albrechts-University Kiel, Computer Science Institute, 24098 Kiel, Germany
thalheim@is.informatik.uni-kiel.de
http://www.is.informatik.uni-kiel.de/∼ thalheim

Abstract. Modelling is one of the central activities in Computer Science. Models are used as intermediate artifacts for system construction, as reasoning tools, for explanation, for description of reality, and for prediction of system behaviour. In this paper we introduce a novel conception of model. The range of models spans from elementary to matured models. Depending on the goal, purpose and function of the model within a deployment story ('Gebrauchsspiel') we may use the most appropriate model. If goals, purposes and functions are changing then we need to change the model.

Keywords: model, models in science, model theory, conceptual model, conception of model.

1 Introduction

It is common misbelief (e.g., [2, 13] or more generally almost all Computer Science textbooks) that there is **no** definition of the conception of the model. We consider this claim as the *big misunderstanding* of the science and art of modelling.

1.1 A First Conception of the Model

As a starting point, a model can be simply considered to be a material or virtual *artifact* (1) which is called a model within a community of practice (2) based on a judgement (3) [8] of appropriateness for representation of other artifacts (things in reality, systems, ...) and serving a *purpose* (4) within this community. We distill in the sequel criteria for artifacts to become a model. We can use two approaches: abstract properties and criteria for artifacts. We observe however that most properties of models are hidden or implicit and thus not given. The user must be a member of a community and base his/her understanding on the background accepted in this community. This implicitness is the main source of misunderstandings on models.

Additionally models are used in many different deployment scenarios and stories ('Gebrauchsspiel' (deployment story), 'Sprachspiel' (language game)) [29]). They are used for certain purposes and have a function within their deployment [9, 18, 30]. Often they should not or cannot be used outside their purpose.

W. Abramowicz (Ed.): BIS 2013, LNBIP 157, pp. 113–124, 2013.

1.2 The Manifold of Model Definitions

Computer Science uses more than 50 different kinds of models in all its sub-disciplines [25]. Business informatics mainly uses four approaches to modelling [26]:

1. The general model definition by Stachowiak [17] uses the mapping, truncation and pragmatic properties for origins and models
2. The axiomatic model definition [1] based on mathematical logics uses formal systems and formal theories which models represent a certain part of reality.
3. The mapping-based model definition [10] uses a direct homomorphic mapping between origin and model for description and prescription and another mapping between model and implemented system for realisation.
4. The construction-oriented model definition [3, 12, 19, 30] considers the model as a result of a modelling process by some community of practice.

These differences explain the variety of model definitions used in literature ([26] selected 35 of notions that are commonly used in business informatics.). As a very short list we may consider the following statements:
[1]: A model is a mathematical description of a business problem.
[3]: A model is the result of a construction process for which the selected part of the origin satisfies the purpose.
[4]: A model is the representation of an object system for the purpose of some subject. It is the result of a construction process by the modeller who addresses a representation of these objects for model user at a certain time and based on some language. A model consists of this construction, the origin, the time and a language.
[8]: A model can be simply considered to be a material or virtual artifact which is called model within a community of practice based on a judgement of appropriateness for representation of other artifacts (things in reality, systems, ...) and serving a purpose within this community.
[11]: The model prescribes concepts as a particular kind of relation relating a subject and an entity.
[15]: A model is an object which has been developed and is used for solution of tasks which cannot be directly solved for the origin by a subject because of its structural and behavioural analogy to an origin.
[18]: Models are governed by the purpose, are mappings of an origin and reflect some of the properties observed or envisioned for the origin. They use languages as carrier.

1.3 Brief Survey of the Paper

Concepts [28] are used for classification or are all the knowledge that the person has, and associates with, the concept's name. *Conceptions* [28] are however systems of explanation. We target in this paper at the conception of the model in Computer and other Sciences.

2 Models Defined through Properties, Purposes and Functions

2.1 Stachowiak, Aristoteles, Galilei and Mahr Properties of Models

Models are often defined through abstract properties they must satisfy [22, 23].

(1) *Mapping* property: Each model has an origin and is based on a mapping from the origin to the artifact.
(2) *Truncation* property: The model lacks some of the ascriptions made to the original and thus functions as an Aristolean model by abstraction of irrelevant.
(3) *Pragmatic* property: The model use is only justified for particular model users, tools of investigation, and period of time.
(4) *Amplification* property: Models use specific extensions which are not observed for the original.
(5) *Distortion* property: Models are developed for improving the physical world or for inclusion of visions of better reality, e.g. for construction via transformation or in Galilean models.
(6) *Idealisation* property: Modelling abstracts from reality by scoping the model to the ideal state of affairs.
(7) *Carrier* property: Models use languages and are thus restricted by the expressive power of these languages.
(8) *Added value* property: Models provide a value or benefit based on their utility, capability and quality characteristics.
(9) *Purpose* property: Models and conceptual models are governed by the purpose. The model preserves the purpose.

The first three properties have been introduced by Stachowiak [17]. The fourth and fifth property have been introduced by Steinmüller [18]. The seventh property is discussed by Mahr [10]. The sixth, eight and ninth properties [23] are however equally if not the most important ones.

2.2 The Purpose of a Model

The purpose dimension is *ruling* and *governing* the model, the development process and the application process because of the main reason for using a model is to provide a solution to a problem. Therefore the purpose is characterised by the solution to the problem provided by the model. We may distinguish a number of concerns such as

the impact of the model *("whereto")* for a solution to a problem,
the insight into the origin's properties *("how")* by giving details how the world is structured or should be structured and how the functionality can be described,
restrictions on applicability and validity *("when")* of a model for some specific solutions, for the validity interval, and the lifespan of a model,
providing reasons for model value *("why")* such as correctness, generality, usefulness, comprehensibility, and novelty, and
the description of functioning of a model *("for which reason")* based on the model capacity.

Purpose is often defined via intention and mixed with function. *Goal* (or intention or target or aim) is a ternary relation between a current state, envisioned states, and people (community of practice). Typical - sometimes rather abstract - intentions are perception support, explanation and demonstration, preparation to an activity, optimisation, hypothesis verification, construction, control, and substitution.

2.3 The Function of a Model while Deploying It in Applications

The *deployment function* of a model relates the model purpose to a practice or application cases or application 'game' similar to Wittgenstein's language game (we call it better *deployment case* and is characterised by answering the classical W-questions (Hermagoras of Temnos[1] rediscovered by J. Zachman): how, when, for which/what or why, at what/which (business use case), etc. We add to purpose: application, conventions, custom, exertion, habit, handling, deployment, service, usage, use, and way of using. The model has a role and plays its behaviour within this application game. [24]:

Description-prescription function: models as images, figures, standard, opus, exposition, representation, composition, realisation;

Explanation function: models 'gestalt', pattern, guidance, type, family or species, original, concept, principle, form, workout;

Optimisation-variation function: models as creation, ideal, achievement, probe, article, plan, variant, substitute;

Verification-validation-testing function: models as sample, schema, specimen, pattern;

Reflection-optimisation function: models as creation, design, construction, type, derivative, master piece, product;

Explorative function: models as result, product, work, art piece, metaphor, paradigm, first edition, style, realisation, artefact;

Hypothetical function: models as copy, release, original form, offshoot, simulation or experiment product;

Documentation-visualisation function: models as presentation, figure, illustration, demonstration, explanation, adornment, plastic, structure.

3 The Conception of Model

3.1 Conceptions to Be Given for a Model

Models are often only considered with their intext, i.e., their structures and behaviour. Context is either neglected or taken for granted. We must however relate a model to the context dimension if we want to understand, deploy or modify the model.

Models follow typically some modelling schemata or pattern [7]. They are based on conceptions (concepts, theoretical statements (axioms, laws, theorems, definitions), models, theories, and tools). Conceptual processes include procedures, conceptual (knowledge) tools and associated norms and rules. Conceptions and conceptual processes are based on paradigms which are corroborated.

Models support interaction, understanding, sharing, and collaboration among people. They depend on existing knowledge, the actual (ontological) state of the reality, the condition of the person's senses and state of mind, and the state of employed instruments. Therefore, models depend on the background concepts that are accepted in a community.

[1] The rhetor Hermagoras of Temnos, as quoted in pseudo-Augustine's De Rhetorica defined seven "circumstances" as the loci of an issue: Quis, quid, quando, ubi, cur, quem ad modum, quibus adminiculis(W^7: Who, what, when, where, why, in what way, by what means). See also Cicero, Thomas Aquinas, and Qunitillian's loci argumentorum as a frame without questioning.

We can summarise the considerations so far and develop a *general model frame*, i.e., a model of the model itself. It consists of four main components

Founding concepts: A model is based on paradigms, background theories, assumptions and guiding principles. It is composed of base conceptions/concepts with a certain scope, expressions, concept space organisation, and some quantification/measurement). Language-based models use a namespace or ontology as a carrier. This namespace is based on definitions made, i.e., the cargo in the sense of [10].

Structure and behaviour: A model is often incrementally built. Models can be multi-faceted (with a specific topology/geometry, with states, with interactions, with causal associations) or monolithic.

Application domain context: A model corresponds to a part of the reality, i.e. the application domain. The domain forms the empirical scope of the model. General or application-specific correspondence rules guide the association between the origin and the model. Each application domain is based on general laws one might have to consider for the model as well.

Meta-model: Models are developed within a theory and have a status within it. These theories provide the content of paradigms. Concepts are the most elementary building blocks. The construction process of a model is guided by the laws applicable to such theory. We may use basic models, emergent models, and subsidiary models.

Reusing the theories of concepts, content and topics [20], we shape the *general concept frame*. A concept is given by the scope, by at least one expression, by its association to other concepts and its media type [14] for the content. The application domain and potential functions constitute the scope of a concept. A concept can be defined by one or more partially synonymous expressions in a definition frame [22]. The concept space must follow some internal organisation. Concepts are interdependent and associated with each other. A concept must be underpinned and quantified by some data which use a certain format. We assume that the formatting can be given by a media type.

3.2 Model Fitness

[7] introduces *model viability*. We extend this approach and consider fitness of a model. Fitness (or superior quality) of a model is given by

(a) *usability* of the model for its purpose, i.e., for resolving the questions, e.g., *validity* of the model;
(b) *potential* of the model for the purpose, i.e., for the goals that are satisfied by the model, e.g., *reliability* and *degree of precision* of the the model;
(c) *efficiency* of the model for the function of the model within the application, i.e., the practice [29] of deployment of the model;
(d) *generality* of the model beside its direct intention of construction of the model, i.e., for applying the model to other goals or purposes, within another function or with some modification or extension, e.g., the extend of *coverage* in the real world.

These four criteria form main quality characterisations of a model. Viability is defined through validity, reliability for the model purpose and function, extent of coverage in dependence on context such as space and time, and efficiency of the model. Viability thus can be used to evaluate how well the model represents the reality for a given scope and how suitable or instrumental is the model for its purpose and function.

The *potential* of a model is defined by its strengths, weaknesses, opportunities and threats (SWOT). The potential can be assessed within a SWOT analysis. A model must be empirically corroborated in dependence on the objective. The abstraction property [17] determines the degree of corroboration. A model must be consistent with the context and the background and coherent in its construction. Models are parsimonious reductions of their origins. Due to this reduction models must be revisable for changes in reality. At the same time models must be relatively stable and robust against minor changes.

3.3 The Background of a Model

To summarise, the background of a model thus consists of

grounding $\mathcal{G}$, i.e., concepts, foundations, language as carrier, and the cargo,
(meta-)basis $\mathcal{B}$, i.e., basement, paradigms, theories; status in application; context; paradigmatic evolution; abstraction, and scale,
deployment $\mathcal{D}$, i.e., goal, purpose, function, and reason,
community of practice $\mathcal{P}$, i.e., stakeholder with their roles and plays, with their interests, portfolio and profiles,
context $\mathcal{C}$, i.e., time, space, and scope,
quality $\mathcal{Q}$, i.e., correctness, generality, usefulness, comprehensibility, parsimony, robustness, novelty etc. and
viability $\mathcal{V}$, i.e., corroboration, coherence, falsifiability, stability, and assurances (restrictions, modality, and confidence).

The grounding and the basis are metaphorically displayed as the cellar and the fundament in Figure 1. The context and the community of practice are two of governing directives for the model.

3.4 Properties of Artifacts Which Might become Models

We can now classify artifacts:

Well-founded artifacts also explicitly define their grounding and basis.
Goal-oriented artifacts describe the goal state(s) and the initial state together with a criterion in which case the goal has been achieved.
Purposeful artifacts are goal-oriented artifacts that provide methods for achieving the purpose.
Deployable artifacts can be applied in deployment stories depending on the artifact orientation (which application story, function).
Useful artifacts also consist of utilisation methods depending on the focus (what is going to be represented, capacity), scope (what shall be solved by the model, purpose) and orientation (which application story, function).
Adequate artifacts $\mathcal{M}^*$ satisfy an invariance property for the origins $\mathcal{M}^*, \mathcal{M}_1, ..., \mathcal{M}_k$ depending on focus and scope.
Homomorphic artifacts are homomorphic to their origins based on some homomorphism.

Conceptual artifacts contain concepts which are used as the basis of semantics for elements of the artifact.
Contextual artifacts explicitly describe their context, e.g., application domain, time, space, discipline, and understanding within a school.
Manufacturable artifacts can be (re-)produced by application of creation methods.
Characterising artifacts describe certain properties of origins $\mathcal{M}^*, \mathcal{M}_1, ..., \mathcal{M}_k$.
Viable artifacts are corroborated, justified and established.
Evaluated artifacts are given with their quality properties.
Supported artifacts are explicitly supported by some community of practice.

3.5 An Artifact Which Is a Model

Given a collection of artifacts $\mathcal{M}^*, \mathcal{M}_1, ..., \mathcal{M}_k$, a community of practice $\mathfrak{P}$ ($\subseteq \mathcal{P}$), a grounding $\mathfrak{G}$ ($\subseteq \mathcal{G}$), viability $\mathfrak{V}$ ($\subseteq \mathcal{V}$), bases $\mathfrak{B}$ ($\subseteq \mathcal{B}$), context $\mathfrak{C}$ ($\subseteq \mathcal{C}$), and deployment $\mathfrak{D}$ ($\subseteq \mathcal{D}$). The adequacy, fitness and usefulness can be expressed through quality characteristics or measures: $\mathfrak{Q}_a \cup \mathfrak{Q}_f \cup \mathfrak{Q}_u$ for adequacy, fitness, and usefulness. We call the quintuple $(\mathfrak{P}, \mathfrak{G}, \mathfrak{V}, \mathfrak{B}, \mathfrak{C}, \mathfrak{D})$ judgement frame $\mathfrak{J}$.

Based on $\mathfrak{J}$ the artifact $\mathcal{M}^*$ is called a **model**

- for $\mathcal{M}_1, ..., \mathcal{M}_k$ by $\mathfrak{P}$,
- for $\mathfrak{D}$ with $\mathfrak{V}$
- if it is appropriate with $\mathfrak{Q}_a \cup \mathfrak{Q}_f \cup \mathfrak{Q}_u$ and within $\mathfrak{C}$
- based on $\mathfrak{B}$ using $\mathfrak{G}$, i.e.,
  - it is adequate (has potential for goals) [similar + regular + fruitful + simple] according to the relation between $\mathcal{M}^*$ and $\mathcal{M}_1, ..., \mathcal{M}_k$ at level of $\mathfrak{Q}_a$ for $\mathfrak{D}$ with $\mathfrak{V}$ within $\mathfrak{B}$ and grounded by $\mathfrak{G}$,
  - it is fit for $\mathfrak{D}$ with $\mathfrak{Q}_f$ within $\mathfrak{C}$ and compliant with $\mathfrak{G}$ and $\mathfrak{B}$ and
  - it is useful for $\mathfrak{P}$ within their $\mathfrak{D}$ and at level of $\mathfrak{Q}_u$.

We may also consider a model suite [6, 21] $\mathcal{M}_1^*, ..., \mathcal{M}_n^*$ instead of $\mathcal{M}^*$.

Therefore, artifacts such as metaphors, parable, similes, allegories, code or programs without additional artifacts for documentation, sets of examples, simple artifacts without any utilisation method, artifacts without deployment, demonstration samples, etc. are *not* considered to be models. Also, the so-called models in data mining are not yet matured to become a pre-model.

3.6 Matured and Elementary Models

A *matured* model is an artifact that uses

a fundament with
- the grounding and
- the (meta-)basis,

four governing directives given by
- the artifacts to be represented by the model,
- the deployment or profile of the model such as goal, purpose or functions,
- the community of practice acting in different roles on certain rights through some obligations, and
- the context of time, discipline, application and scientific school,

two pillars which provide

- methods for development of the model and
- methods for utilisation of the model,

and finally

the model portfolio and function for the deployment of the model in the given application.

An *elementary* model is an artifact that

- represent artifacts,
- has a deployment or profile of the model such as goal, purpose or functions,
- is supported by a (partial) community of practice, and
- has (some) methods for utilisation of the model.

A **model** is thus an artifact that consists of an elementary model and that may be extended to a matured model.

The *model house* in Figure 1 displays these different facets of the model. It displays the matured or fully fledged model with grounding and basis as the fundament,

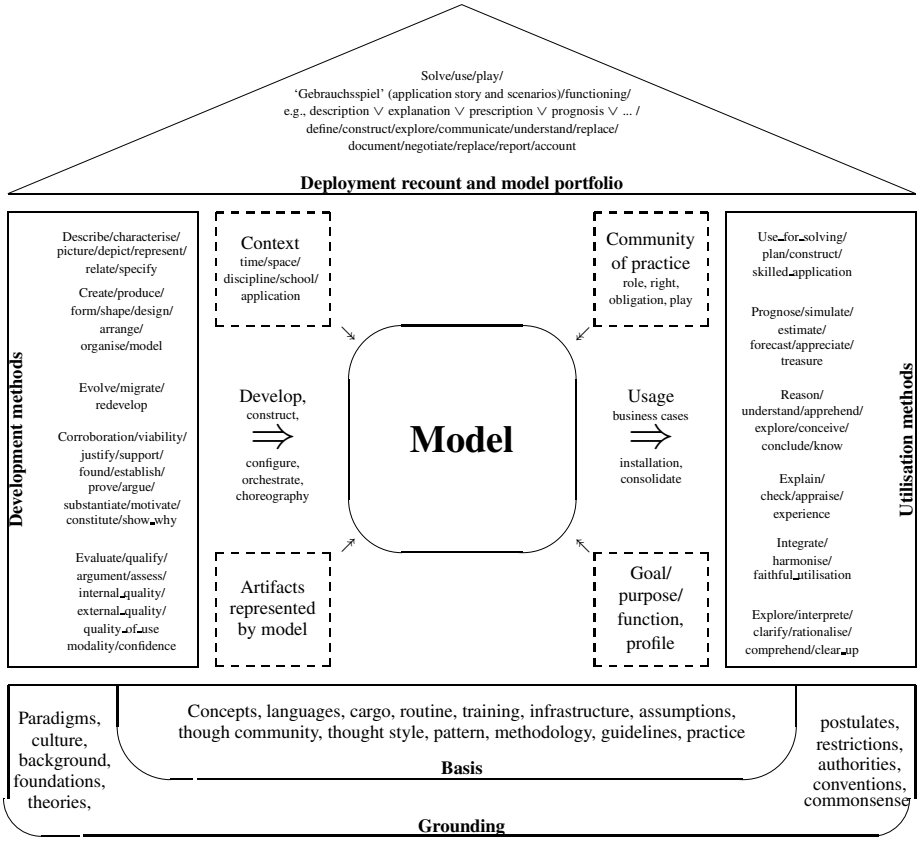


Fig. 1. The matured model

with four governing directives, with technical and technological pillars for development and utilisation, and with the application roof The house consists of a cellar and a fundament, two pillars, four driving or governing forces, and finally the deployment roof. The *grounding* is typically implicitly assumed. It contains paradigms, the culture in the given application area, the background, foundations and theories in the discipline, postulates, (juristical and other) restrictions, conventions, and the commonsense. The *basis* is the main part of the background. It is typically given for modelling. The development uses a variety of methods for *description*, *construction*, *evolution*, *corroboration*, and *evaluation*. The utilisation is based on methods for *applying*, *prognosis*, *reasoning*, *explanation*, and *exploration*. We have used different verbs for classification of the activities. The model can be used for completion of certain tasks. These tasks may be combined into a *model portfolio*. The model is used for certain functions or 'Gebrauchsspiel' (application story or 'game'). Finally, the model is *governed* by four directives: *artifacts*, *profile*, *community of practice*, and *context*.
The nine properties of models can now naturally be represented:

The **mapping property** is one kind of model association.
The **truncation property** concentrates on abstraction as some kind of association.
The **pragmatic property** is given by the goal, community, and the context (time).
The **extension property** uses a specific partiality of mapping into instead of being surjective.
The **distortion property** is a specific kind of mapping related to the goal.
The **idealisation property** is another specific kind of mapping.
The **carrier property** relates the model to its grounding and basis.
The **added value property** is one kind of combined quality.
The **purpose property** is a more explicit part of the pragmatic property.

3.7 Pre-models

Pre-models are artifacts which do not contain an elementary model. In literature they are often called "models". They leave however out many essential parts and can thus be misinterpreted, misunderstood, and misleading. The conclusions drawn with such models are often doubtful or spurious.

Metaphors, similes and allegories are often considered to be models. They are however presented without a clear grounding, context or profile. The artifacts are often incomplete. There development methods are left out. The community of practice is only partially given. Furthermore, the basis is partially given. This situation can also be observed in other disciplines.

Models might have a profile that consists only of goals. Methods for development and utilisation are thus not necessary. The deployment is then vague. Illustrations are typical models of the restricted applicability.

3.8 Elementary Models and Elementary Pre-models

Computer Science models often do not discuss the basis or grounding. For instance, the basis of Turing machines includes a number of implicit principles such as compositionality, functional or relational state transformations, step-wise computation, and context-freeness. One of the guiding implicit postulates is the Von-Neumann-machine and its

sequentiality. The purpose of the Turing machine model is a theory of computability and complexity. Construction of real machines is beyond its purpose. Instead, the abstract state machine approach explicitly uses the three guiding postulates for sequential computation (postulates of sequential time, of abstract state, of bounded exploration of the state space)[5]. These postulates may be extended to postulates for parallel and concurrent computation, e.g., by extending the last postulate to the postulate of finite exploration.

Mathematical models are often built around elementary models[2]. The main interest of these models is utilisation and partially deployment. A systematic and well-founded development frame has been presented in [16]. [27] discusses an evaluation frame and a guide to assessment of mathematical models. A similar observation can be made for missing profiles, e.g., goals, purposes, or functioning.

Figure 2 depicts the relationship of pre-models and elementary (pre-)models to the matured model[3]. Some mathematical models might also be pre-models. Pre-models can become mathematical models if at least the profile is given.

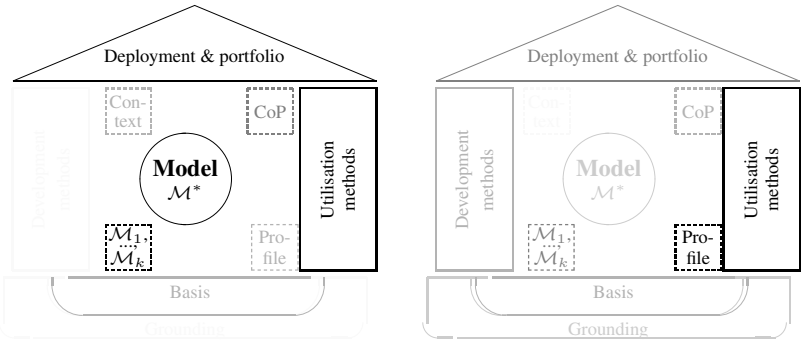


Fig. 2. Pre-models and elementary models compared to matured models in Figure 1

4 Conclusion

In this paper we introduced the conception of a model. This paper completes [22–24] by an explicit definition of the conception of the model, by separation of concern within the 'model house', and by explicit exclusion criteria for artifacts that can not be considered to be models. This conception has been applied and tested in Computer Science, Philosophy, Physics, and other sciences. So far we discovered that the notion is sufficient

[2] The classification of mathematical models seems to be very rigid. Development methods are typically not a matter for mathematicians. The basis and the grounding are of very partial interest. The classical scenario in mathematical modelling is: receive a model and application problems; transform the model and one of the problems to a system of mathematical notions; analyse, simulate and solve the selected problem; interpret the solutions within the received model and problems.

[3] We use the gray colour for explicit presentation of rudimentary elements of these models. Missing parts are not shown at all.

and necessary and thus serves requirements for becoming a definition. It has been not our aim to define a complete formal theory. Each term used in the paper needs however such formal underpinning. The formal theory is however a straightforward application of Discrete Mathematics and thus not a goal for a conference paper. The description and the theory of methods used either for development of models or for utilisation of models is a challenging research issue and cannot be handled yet.

We have been aiming at development of a formal notion of a model. Such formal notion is necessary whenever we need a theory of modelling. It allows to exclude artifacts to become a model outside the judgement frame. It allows also to state when an artifact is a model and what is necessary for an artifact to become a model.

Acknowledgment. We would like to thank the colleagues from the Christian-Albrechts University at Kiel for the fruitful discussions on many facets of models in archeology, arts, biology, chemistry, computer science, economics, electrotechnics, environmental sciences, farming, geosciences, historical sciences, languages, mathematics, medicine, ocean sciences, pedagogical science, philosophy, physics, political sciences, sociology, and sports. We are thankful to the International Institute of Theoretical Cardiology (IIfTC) for the evaluation of our approach. These discussions lasted in weekly 'Tuesday' open-end-evening seminars over the last three years from 2009 until now and in monthly seminars at the IIfTC. They resulted in a general understanding of the notion of a model in most sciences, engineering and technology.

References

1. Abts, D., Mülder, W.: Grundkurs Wirtschaftsinformatik: Eine kompakte und praxisorientierte Einführung. Vieweg (2004)
2. Agassi, J.: Why there is no theory of models. In: Niiniluoto, I., Herfel, W.E., Krajewsky, W., Wojcicki, R. (eds.) Theories and Models in Scientific Processes, pp. 17–26. Amsterdam-Atlanta (1995)
3. Alpar, P.: Computergestützte interaktive Methodenauswahl. PhD thesis, Frankfurt Main Univ. (1980)
4. Becker, J., Schütte, R.: Handelsinformationssysteme: Domänenorientierte Einführung in die Wirtschaftsinformatik. Moderne Industrie (2004)
5. Börger, E., Stärk, R.: Abstract state machines - A method for high-level system design and analysis. Springer, Berlin (2003)
6. Dahanayake, A., Thalheim, B.: Co-evolution of (Information) system models. In: Bider, I., Halpin, T., Krogstie, J., Nurcan, S., Proper, E., Schmidt, R., Ukor, R. (eds.) BPMDS 2010 and EMMSAD 2010. LNBIP, vol. 50, pp. 314–326. Springer, Heidelberg (2010)
7. Halloun, I.A.: Modeling Theory in Science Education. Springer, Berlin (2006)
8. Kaschek, R.: Konzeptionelle Modellierung. PhD thesis, University Klagenfurt, Habilitationsschrift (2003)
9. Loos, P., Fettke, P., Weißenberger, B.E., Zelewski, S., Heinzl, A., Frank, U., Iivari, J.: Welche Rolle spielen eigentlich stilisierte Fakten in der Grundlagenforschung der Wirtschaftsinformatik? Wirtschaftsinformatik 53(2), 109–121 (2011)
10. Mahr, B.: Information science and the logic of models. Software and System Modeling 8(3), 365–383 (2009)
11. Mahr, B.: Intentionality and modeling of conception. In: Judgements and Propositions. Logical, Linguistic, and Cognitive Issues, Logos (2010)

12. Ortner, E.: Melchios - Methodenneutrale Konstruktionssprache für Informationssysteme. Technical Report Bericht 60-94, Universität Konstanz, Informationswissenschaft (1994)
13. Ritchey, T.: Outline for a morphology of modelling methods -Contribution to a general theory of modelling. Acta Morphologica Generalis 1(1), 1–20 (2012)
14. Schewe, K.-D., Thalheim, B.: Usage-based storyboarding for web information systems. Technical Report 2006-13, Christian Albrechts University Kiel, Institute of Computer Science and Applied Mathematics, Kiel (2006)
15. Scholz-Reiter, B.: Konzeption eines rechnergestützten Werkzeugs zur Analyse und Modellierung integrierter Informations- und Kommunikationssysteme in Produktionsunternehmen. PhD thesis, TU Berlin, Informatik (1990)
16. Sovetov, B.J., Jakovlev, S.A.: Systems Modelling. Vysschaja Schkola (2005) (in Russian)
17. Stachowiak, H.: Modell. In: Seiffert, H., Radnitzky, G. (eds.) Handlexikon zur Wissenschaftstheorie, pp. 219–222. Deutscher Taschenbuch Verlag GmbH & Co. KG, München (1992)
18. Steinmüller, W.: Informationstechnologie und Gesellschaft: Einführung in die Angewandte Informatik. Wissenschaftliche Buchgesellschaft, Darmstadt (1993)
19. Thalheim, B.: Entity-relationship modeling – Foundations of database technology. Springer, Berlin (2000)
20. Thalheim, B.: The conceptual framework to user-oriented content management. In: Information Modelling and Knowledge Bases, XVII. Series Frontiers in Arificial Intelligence, vol. 154, pp. 30–49 (2007)
21. Thalheim, B.: Model suites for multi-layered database modelling. In: Information Modelling and Knowledge Bases XXI. Frontiers in Artificial Intelligence and Applications, vol. 206, pp. 116–134. IOS Press (2010)
22. Thalheim, B.: Towards a theory of conceptual modelling. Journal of Universal Computer Science 16(20), 3102–3137 (2010), http://www.jucs.org/jucs_16_20/towards_a_theory_of
23. Thalheim, B.: The theory of conceptual models, the theory of conceptual modelling and foundations of conceptual modelling. In: The Handbook of Conceptual Modeling: Its Usage and Its Challenges, ch. 17, pp. 547–580. Springer, Berlin (2011)
24. Thalheim, B.: The science and art of conceptual modelling. In: Hameurlain, A., Küng, J., Wagner, R., Liddle, S.W., Schewe, K.-D., Zhou, X. (eds.) TLDKS VI. LNCS, vol. 7600, pp. 76–105. Springer, Heidelberg (2012)
25. Thomas, M.: Modelle in der Fachsprache der Informatik. Untersuchung von Vorlesungsskripten aus der Kerninformatik. In: DDI. LNI, vol. 22, pp. 99–108. GI (2002)
26. Thomas, O.: Das Modellverständnis in der Wirtschaftsinformatik: Historie, Literaturanalyse und Begriffsexplikation. Technical Report Heft 184, Institut für Wirtschaftsinformatik, DFKI, Saarbrücken (Mai 2005)
27. von Dresky, C., Gasser, I., Ortlieb, C.P., Günzel, S.: Mathematische Modellierung: Eine Einführung in zwölf Fallstudien. Vieweg (2009)
28. White, R.T.: Commentary: Conceptual and conceptional change. Learning and Instruction 4, 117–121 (1994)
29. Wittgenstein, L.: Philosophical Investigations. Basil Blackwell, Oxford (1958)
30. Zelewski, S.: Kann Wissenschaftstheorie behilflich für die Publikationspraxis sein? In: Lehner, F., Zelewski, S. (eds.) Wissenschaftstheoretische Fundierung und Wissenschaftliche Orientierung der Wirtschaftsinformatik, pp. 71–120. GTO (2007)

Using Markov Decision Process for Recommendations Based on Aggregated Decision Data Models

Razvan Petrusel

Faculty of Economical Sciences and Business Administration, Babes-Bolyai University,
Teodor Mihali str. 58-60, 400591 Cluj-Napoca, Romania
razvan.petrusel@econ.ubbcluj.ro

Abstract. Our research is placed in the context of business decision making processes. We look at decision making as at a workflow of (mostly mental) activities directed at choosing one decision alternative. Our goal is to direct the flow of decision activities such that the relevant alternatives are properly evaluated. It is outside our purpose to recommend which alternative should be chosen. Since business decision making is data-centric, we use a Decision Data Model (DDM). It is automatically mined from a log containing the decision maker's actions while interacting with business software. The recommendation is based on an aggregated DDM that shows what many decision makers have done in the same decision situation. In our previous work we created algorithms that seek a local optimum. In this paper we show how the recommendation based on DDM problem can be mapped to a Markov Decision Process (MDP). The aim is to use MDP to find a global optimal decision making strategy.

Keywords: Decision Process Recommendation, Decision Data Model, Markov Decision Process.

1 Introduction

Decision making is the essential concern for any business manager. Our approach looks at decision making as at a workflow (set of actions, sequence, and concurrence) of (mostly) mental actions aimed at choosing a decision alternative. This is an approach different from the ones in classical decision theory, which blends the process perspective with the data view. For example, in order to choose to buy or not to buy some stock, the stock market investor will look at the daily values of that stock, calculate the trend, compare it with some other stock, etc. Since the quality of the decision (choice) is directly influenced by the underlying decision process [1], we look at the actual activities performed by an individual decision maker and at their sequence. For the previous example it is obvious that, if the decision maker didn't consider checking upcoming regulatory laws, it is likely that the choice of whether to buy or not would be flawed.

But, how could a decision maker know which actions can improve the decision outcome? Well, if his own knowledge or experience didn't suggest it, then somebody else should. This is where our approach comes in handy. We log the decision actions of many individuals while making the same decision and we mine an aggregated

W. Abramowicz (Ed.): BIS 2013, LNBIP 157, pp. 125–137, 2013.

model. Then, based on the aggregated model, we provide some recommendations of the next action that should be performed, given a state of the current process. The basic assumption is that if many decision makers performed some action, and that action was not performed in the current process, it should be recommended. This approach is tailored towards data-centric business decision making, and an action is seen as some manipulation of data.

The goal of our research is to help a person, faced with the necessity to make a certain decision, make a better informed decision. By providing recommendations, we try to guide the process so that all relevant criterions are considered and the alternatives are properly evaluated. We do not aim to automate decision making or diminish the role of the decision maker.

The work presented in this paper builds on the Process Data Model (PDM) recommendation based on Markov Decision Process (MDP) of Vanderfeesten[2]. It is also a follow-up of our own work that aims at producing a complete framework for decision-process knowledge extraction and modeling [3], [4], [5]. Our framework already includes a Greedy and an A* based approach that provide a local optimal recommendation. The MDP is employed in problems where a global optimum is needed. Therefore, the motivation of this paper is to improve on our previous work by providing a decision maker with a global optimal decision making strategy.

The next section of the paper introduces the context of our research. The third section lays the formal foundation. The fourth section we introduce the algorithm and in the next section we briefly discuss our preliminary findings. The last two sections deal with the related work and the conclusions.

2 Overview

This section introduces an overview of our previous work in order to provide the context of the paper. This paper focuses on a small part of this framework (i.e. providing recommendations based on a previously mined decision model).

We look at business decision making processes as at a data-centric environment. Most of the decision making in this environment evolves on using and interpreting data and building information based on that. But, creating information from data is not a trivial task while it is essential for the outcome of a business decision. Therefore, the entire framework aims to extract and make explicit the activities related to data and information manipulation.

To gain insights into how a decision maker manipulated data, the obvious approach is to perform an interview. This leads to knowledge representation under different formalisms (rules, decision trees, influence diagrams, etc.). If, for various reasons, one expert is not enough, more interviews need to be carried out and the output models need to be updated to show the aggregate view. The problem with this approach is that is quite difficult and expensive to perform interviews with many individuals and troublesome to update models with extra behavior, so that a broader and more reliable model is created.

Our approach takes a different path. We argue that a model, comparable with the one extracted from several experts, can be created if we mine the behavior of a large

number of decision makers. We created a framework (see Fig. 1) that allows us to extract a model using this assumption. We argue that the decision behavior is made explicit by the footmarks left by the users of a decision-aware software. The main features of such software are as follows: a) it shows the raw data for the decision; b) offers the tools to manipulate it; and c) logs everything the user does. One can think, for example, to an Excel Spreadsheet in which the main figures about a company are shown while the user needs to make a decision such as contracting a loan. The software offers the tools to manipulate data (by formulas or functions) and it can be enhanced to log what the user does. Logging can be either low-level (e.g. clicks on cells, formulas used in cells) or high-level (e.g. performing eye-tracking of the user). We look at the logged behavior of a decision maker as at a trace of the decision process.

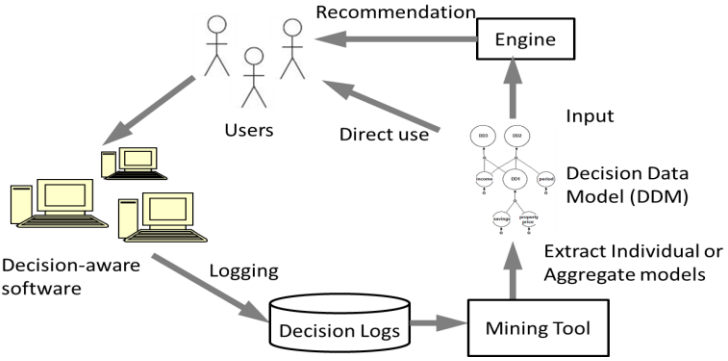


Fig. 1. The framework of decision making process mining

Once the decision logs are available, we need some algorithm to mine the model. In [5] we showed that current process mining algorithms do not perform well on such decision logs. Therefore, in [3] we proposed a model (DDM – Decision Data Model), derived from the Product Data Model [6], that is better suited for such a field. In [5] we showed how multiple individual DDMs can be aggregated. A DDM can be easily read by humans, but it could also be used as input in a recommendation engine. In [4] we introduced two approaches to recommendations: one based on a Greedy approach and one derived from the A* path-finding algorithm. This paper introduces the third algorithm that uses an aggregated DDM as input. It is mapped to a Markov Decision Process and, unlike the other two algorithms that recommended the local next best action, provides recommendations for the optimal decision strategy that may be used.

3 Recommendation Based on Markov Decision Process

The statement of the problem we try to tackle is as follows: given a partial decision making trace and an acyclic, not rooted graph, give advice about the 'next *n* actions' that should be performed. 'Next *n* actions' means that there are a finite number of actions that are considered before stopping the decision process and making the

decision. An action needs to be seen as some data manipulation (e.g. dividing two data items to calculate some key figure).

In the remainder of this section we introduce the essential notions used further in the paper. The next sub-section makes the connection with the previous research, the second shows how we can map to the Markov Decision Process while the third introduces a short running example to ease the understanding of the problem.

3.1 Preliminaries

Our problem can be mapped to a general search problem with five components: S, S_0, S_g, *successors* and *cost*, where:

- S is a finite set of states;
- $S_0 \subseteq S$ is the non-empty set of start states;
- $S_g \subseteq S$ is the non-empty set of goal states;
- *successors* is a function $S \rightarrow P(S)$ which takes a state as input and returnsanother state as output (probabilities may be used in connection with it);
- *cost* is a value associated to moving from state $s \in S$ to $s' \in S$).

The total cost is the sum of the *costs* incurred by a sequence of movements from state $s \in S_0$ to a state $s' \in S_g$. A recommended strategy is a sequence of actions such as the total cost is minimized (or maximized under some circumstances). A particular feature of our problem is that, because an action is performed within software, (usually) there are no costs for moving from a state to the next one (as in classical search problems). Instead, the notion of cost is derived from the notion of frequency. Since we want to guide the user through the most frequent path, the cost of an action should be lower if it was performed by many other decision makers and higher if it was performed by just a few other persons. Therefore, the MDP recommendation algorithm will look for a minimal total cost.

Definition 1 (Decision Data Model): The entire approach relies on the Decision Data Model (DDM). It is derived from the Product Data Model (PDM) introduced in [6].A DDM is a tuple (D, O) where D is the set of data elements d, D = BD $\cup$ DD(BD is the set of basic data elements and DD is the set of derived data elements); and O is the set of operations on the data elements (more details in [3]). D and O form a hyper-graph, H = (D, O), connected and acyclic.

Definition 2 (Aggregated DDM): An *aggregated DDM* is a DDM annotated with the *frequency* of the operations. *Frequency* indicates how many times an operation shows up in the log.

Definition 3 (Enabled operation): An operationo is *enabled* when the input data elements (DS) are available (known to the decision maker). If an operation is enabled it may be executed so that the output element d is produced and the output value v is known. When any operation is executed, the process moves from one state to another.

Property1 (Basic operations): All the basic data elements have as input the empty set. Therefore, at the start of the decision process, the only enabled operations are the ones

producing the basic data elements. Executing such operations may be seen as the stage in any decision process in which the decision maker finds out the specific data.

Definition 4 (State of DDM): A *state of a DDM* is a particular distribution of operations over the sets of Enabled, Not-Enabled and Executed operations (see more details in sub-section 3.3). As the process progresses, the different states are represented by different placements of the operations in the three sets. There is no need to reach the end state (when Enabled is empty) in order to make the decision.

3.2 Mapping the Markov Decision Process to DDMs

A MDP is represented by a tuple (S, T, A, P, c, q) where:

- S is the finite state space;
- T is the set of discrete time points with a finite horizon;
- A is the finite decision space;
- P is the transition function such that $p_{ij}(a)$ is the probability that decision a in state i leads to state j at the next time point;
- c is the immediate cost function;
- q is the final cost function when the process finishes [7].

A decision process represented as a DDM can be mapped to an MDP for *S*, *T*, *A*(see the example in the nextsub-section).

Mapping *c* is useless, since there are no costs for performing an operation because we are looking at data processing in software. We substitute the cost of an operation, $c(o_i)$, with the value that is calculated by dividing the sum of occurrences of all operations in the DDM to the number of occurrences, $fq(o_i)$, of a particular operation. Considering this substitution, the goal of our algorithm isto minimize the total cost of a sequence of operations. The formula that assigns a cost to an operation is:

$$c(o_i) = \frac{\sum_{j=1}^{n} fq(o_j)}{fq(o_i)}$$

The probability for moving to a new state is computed based on thefrequencyof all operations that are enabled in one state. The formula that calculates the probability,$p(o_i)$, of performing operation o_i, given a set of enabled operations, *Enabled* = $\{o_k \mid k=1,n\}$ is:

$$p(oi) = \frac{fq(oi)}{\sum_{k=1}^{n} fq(o_k)}$$

We try to find an optimal decision strategy. A decision strategy is a sequence of n operations that can be performed by the decision maker(where *n* is the number of decision epochs). A decision epoch is the time between the occurrences of two sequential operations (between two successive moves from one state to another) [2]. In an epoch, the decision maker thinks about the problem and manipulates some data. When some result is outputted, the epoch ends. The decision epochs are not equidistant. The number of decision epochs is known when the process starts (i.e. the decision maker knows that there is a maximum number of *n* operations that can be performed before he needs to make his decision).

To calculate q, we use the discounted reward notion to model the operation opportunity. Therefore, the cost of an operation in a future decision epoch has a different cost than the cost of the same operation performed in the current epoch. For example, if an operation is performed in the current decision epoch it may cost 100 units while if it is performed in the next decision epoch it will cost 90 units and in the third one 81 units, if we assume a discount factor of 10%. We use discounting to encourage early execution of the most frequent (least expensive) enabled operations.

3.3 MDP Mapping Example

In order to improve the understanding of the aggregate DDM and the algorithm based on it, we will use an example. We mined the aggregated DDM in Fig. 2 from a random selection of 50 traces (the entire log is available at http://www.edirector.ro/v3_1/export/pm.xml). The decision makers had to decide whether to buy or rent a house. The data items that were given to the decision makers are called basic data items (the elements with a name). The items calculated by decision makers (a consequence of performing an operation) are artificially named OUTx, where x is assigned sequentially for each new operation found in the log. Each data item is annotated with its frequency (e.g. OUT1 shows up 32 times in the log). We did not annotate the model with the frequency of operations since it is always equal to the one of the derived data items (in this model there are no derived data items produced by several alternate operations). Also, for simplification, we assume that the label of the derived element is also the label of the operation producing it.

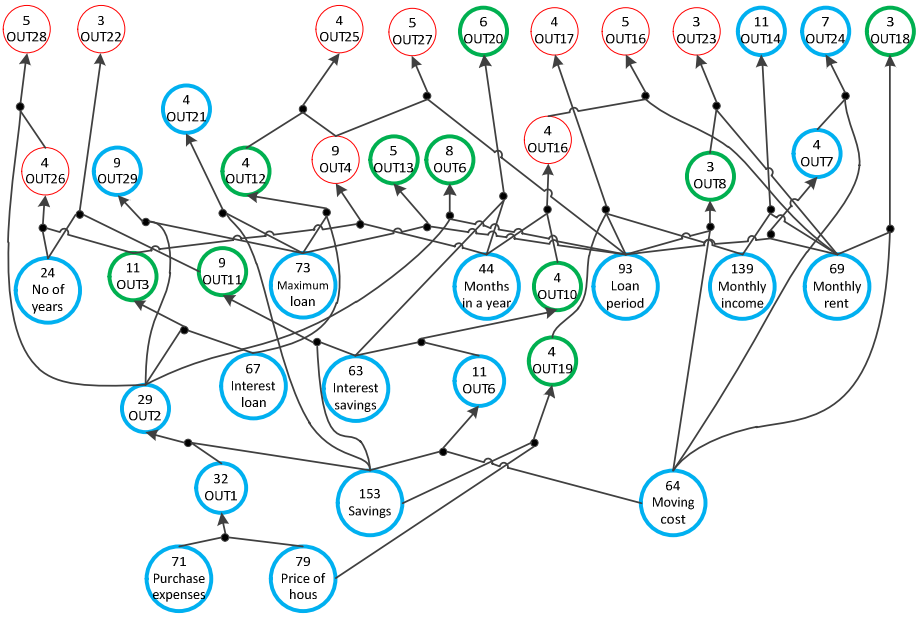


Fig. 2. An Aggregated Decision Data Model

The semantics of the model implies that an operation can be executed only if its input data items are known. For example, OUT2 can be executed only if the values of OUT1 and Savings are both known (i.e. if OUT1 has been calculated previously based on the values of Purchase Expenses and Price of the House.

Since the basic data is readily available (in the software) we assume there is no operation associated with finding it out. It is useless to include in the notion of state this kind of elements. Therefore, the initial state of the DDM is the one where all the derived data (OUT elements) are either Enabled or Not Enabled while Executed is empty.

The mapping of the decision process presented in the DDM in Fig. 2 to a MDP is:

- *S* is the finite number of combinations of distributions of operations over the three sets: Enabled, Executed and Not-Enabled. An example of a state is:

  - Enabled = {OUT3, OUT11, OUT12, OUT13,OUT6, OUT20, OUT19, OUT10, OUT8, OUT18},
  - Executed = {OUT1,OUT2,OUT6,OUT29,OUT21,OUT7,OUT14,OUT24}
  - Not-enabled = {OUT26, OUT4, OUT16, OUT28, OUT22, OUT25, OUT27, OUT17, OUT16, OUT23}.

If, for example, OUT11 would be calculated, the process moves to a new state (i.e. the operation producing OUT11 would becomeExecuted, therefore the one producing OUT22 becomes Enabled):

  - Enabled = {OUT3, OUT12, OUT13,OUT6, OUT20, OUT19, OUT10, OUT8, OUT18, OUT22},
  - Executed = {OUT1, OUT2, OUT6, OUT29, OUT21, OUT7, OUT14, OUT24, OUT11}
  - Not-enabled = {OUT26, OUT4, OUT16, OUT28, OUT25, OUT27, OUT17, OUT16, OUT23}.

- *T* is the time point at which an operation is executed. It is a finite set because the execution of a DDM finishes when all the operations have been executed (since there are no cycles allowed in the model). For example, the initial state in the example occurs at time 8 (i.e. there are 8 executed operations) and the decision to execute the next operation was made at time 9. Therefore, the decision epoch is the time elapsed between time 8 and time 9. The maximum number of remaining decision epochs for the DDM in Fig. 2 is 19 (i.e. it can be completely executed in 28 decision epochs).

- *A* maps to a set of all possible decisions in the DDM. A decision refers to the operation that will be executed. At any given time A = Enabled set. In the example above the change of the process state is triggered by the decision to execute OUT11 while *A* = {OUT3, OUT11, OUT12, OUT13,OUT6, OUT20, OUT19, OUT10, OUT8,OUT18}.We explore what happens if the number of decision epochs is less than the number needed so that the process reaches its end state.

- *P* maps tothe probability to execute an operation given the Enabled set. In the example, at time 8 there are 10 enabled operations: OUT3, OUT11, OUT12, OUT13,OUT6, OUT20, OUT19, OUT10, OUT8,OUT18with frequencies 11, 9, 4, 5, 8, 6, 4, 4 and 3. Therefore, the probabilities for each operation are: 20.38% (11/54), 16.67%, 7.40%, 9.27%, 14.81%, 11.11%, 7.40%, 7.40% and 5.56%. OUT11 was executed at the next decision epoch, therefore, at the next decision time, there are

again10 enabled operations: OUT3, OUT12, OUT13,OUT6, OUT20, OUT19, OUT10, OUT8,OUT18, OUT22 with frequencies 11, 4, 5, 8, 6, 4, 4,3, 3 and probabilities 22.92% (11/46), 8.33%, 10.42%, 16.67%, 12.50%, 8.33%, 8.33%, 6.25% and 6.25%.

-*c* is the cost of each individual operation, which is known before the process starts and does not change during the execution. For example, the cost of OUT11 is 23.33 (i.e. 210, the total frequency of all operations, divided to 9, the frequency of this operation).

- *q* is calculated for each decision epoch as the cost of being in a state (total cost of operations in Executed set) plus the cost of future operations for the remaining decision epochs. For example, the cost of being in state 8 is $c_{OUT1}+c_{OUT2}+c_{OUT6}+c_{OUT29}+c_{OUT21}+c_{OUT7}+c_{OUT14}+c_{OUT24}$= 373.12.

4 The Recommendation Algorithm

This section introduces the algorithm that produces recommendations for a decision process, based on a DDM. We aim to provide the answer to one question: "Which is the decision strategy that provides optimal results, given the decision maker's previous actions?". Therefore, the problem we are faced with can be formalized as: "Select the next *n* operations from the DDM to be performed next, such that the overall cost function would me minimized?".

In any state, the decision maker can choose to execute any operation from the Enabled set, or may perform any new operation that is not in the DDM (as long as the data needed as input for that operation is available). After the decision maker performs an operation (the suggested one or any other), the process moves to a new state and the system provides another recommendation. After each state change the algorithm that is performed is:

1. compute the state space by 'walking' the DDM and the probability for each transition;
2. compute the cost for the next *n* states(i.e. until decision epoch *n*)and for each transition compare the cost with a threshold set a-priori;
3. for each decision epoch*k*, the cost is computed as: the cost of previous states + the discounted cost of future *n-k* reachable states;
4. select min(final_cost(*s*)) and recursively the recommendation sequence. Give as reason for the recommendation the costs for being in each state.

Basically, the algorithm walks all possible states that derive from the current state and iteratively calculates the cost. For each new state the total cost is the cost of getting to the current stateplusthe discounted cost of the future states. We stop exploring further when a new operation adds less than a threshold, to the cost in the previous state.

To better understand how the algorithm works we will use a small running example. We assume we logged several decision traces (and their frequency):op1, op2 (observed 40 times), op1 (15), op1, op2, op3 (35), and op3 (10). Based on this data, we extract: the frequency of each operation (e.g. frequency of op1 is 90). Based on the traces and other dependency information stored in the log, the DDM in Fig. 3 and the state space in Fig. 4 are created.

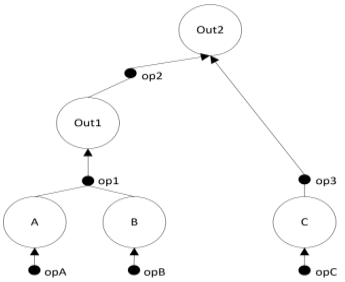


Fig. 3. DDM model used as the running example

In Fig. 4 we show the state space of the DDM in Fig. 3. As shown before, each state is a distribution of operations over the three states. The number in the lower left corner is the aggregated frequency of that particular state while the number in the lower-right corner is the cost of being in that state. The edges show the transitions from one state to another while the labels show the probability of the transition.

One can note that if there are 3 operations, the decision process can finish after 1, 2 or all 3 of the operations are performed (therefore there are 7 possible states). The number of actual states enforced by the DDM is smaller (there are 5 non-trivial states). It is obvious that in the real DDMs the number of states that needs to be explored is considerably less than the number of possible combinations between the operations. Also, in [2] one of the major problems was the state explosion. By combining the restrictions imposed by the DDM structure and the stop condition we seek to reduce the computational effort.

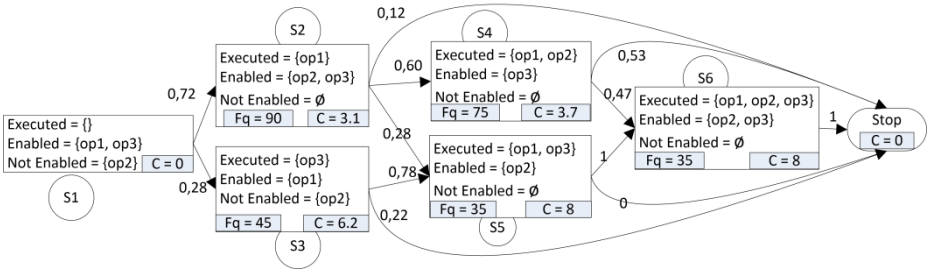


Fig. 4. The state space of the DDM in the running example

From state S1 the process can move either to S2 or S3 (i.e. op1 or op3 from Fig. 3 may be executed) with a probability of 72% (i.e. 90/135) or 28% (i.e. 45/135). From state S1 it is pointless to decide without doing any operation. Then, from state S2 the process can move to states S4, S5 or a decision can be made and the process stops with a probability of 60%, 28% and 12%. One can note that the possible strategies are: 1) "do only op1", 2) "do only op3", 3) "do op1 and op2" and 4) "do op1 and op3". The strategy "do op3 and op2" is not valid because op2 cannot be executed unless op1 is executed previously.

The problem statement for the running example is: "Given a time horizon of 2 decision epochs,a discount rate of 10% per epoch, and the DDM in Fig. 3, give the best decision strategy".

The calculation steps of the algorithm are:

Time 0: $reward_0$ = current cost (S0) +discounted cost(S2) + discounted cost(S3) + $discounted^2$ cost(S2, S4) + $discounted^2$ cost(S2, S5) +$discounted^2$ cost(S2, stop) + $discounted^2$ cost(S3, S5) + $discounted^2$ cost(S3, stop) =0+0,9*72%*3,1+0.9*28% *6.2+0.81*60%*3.7+0.81*28%*8+0.81*12%*0+0.81*78%*8+0.81*22%*0 = 0+2+ 1.56+1.8+1.81+0+5.05+0 = 12.22

- check(S2) =0.9*3.1/(0+0,9*3.1) = 100%> 10% this path will be explored further
- check(S4)=0.9*3.7/(3.1+0.9*3.7)=51.79% >10% this path will be explored further
- ...
- check(S2, stop)=0.9*0/(3.1+0.9*0)=0<10% this path will not be explored further

Time 1:

$reward_{op1}$ =cost(S0) + current cost(S2) + discounted cost(S2, S4) + discounted cost(S2, S5) = 0+3.1+0.9*60%*3.7+0.9*28%*8 = 7.12

$reward_{op3}$ = cost(S0) + current cost(S3) + discounted cost(S3, S5) = 0+6.2+ 0.9*78%*8 = 11.82

Time 2:

$reward_{op12}$ = $reward_0$ +$reward_{op1}$ +current cost(S4) = 0+3.1+3.7 = 6.8

$reward_{op13}$ = $reward_0$ +$reward_{op1}$ +current cost(S5) = 0+3.1+8 = 11.1

$reward_{op31}$ = $reward_0$ +$reward_{op3}$ +current cost(S5) = 0 + 6.2 + 8 = 14.2

Given the example above, we can determine, recursively, for each state the best choice. At time 2 it is obvious that the lowest score is for executing op2. At time 1, the lowest score is for op1. Therefore, the recommendation would be "do op1 and then op2". The reason would be "because it will generate costs of 7.12 and 6.8".

One can note that there are two potential paths that lead to state S5 (i.e. 'op1 then op3' or 'op3 then op1').The one with the lowest cost is 'op1 then op3'.

5 Discussion

This section aims to discuss if the MDP-based approach to Aggregated DDM recommendations, introduced previously, is feasible. Currently, the algorithm is building an initial state space by exploring every sequence of operations allowed in the DDM. This is the most resource consuming operation because a lot of variations need to be checked. For each new operation added to the Enabled set, the cost is discounted and the marginal cost is checked. We stop exploring when, to a path's cost, is added a value with a percentage lower than a fixed threshold. To this regard, we benefit from the fact that an operation that is derived based on another one has a frequency at most equal with the source. Therefore, it is valid to assume that, if the source already contributes very little to the overall score, the dependent discounted operation will contribute with even less. However, setting the threshold and the discounted value is not trivial. So far we experienced with different variations and checked how sensitive the computation time is with regard to the threshold value.

We used the DDM in Fig. 2 to compute the first decision strategy with 20 decision epochs and with various thresholds (see Fig. 5). We arrived to the conclusion that it

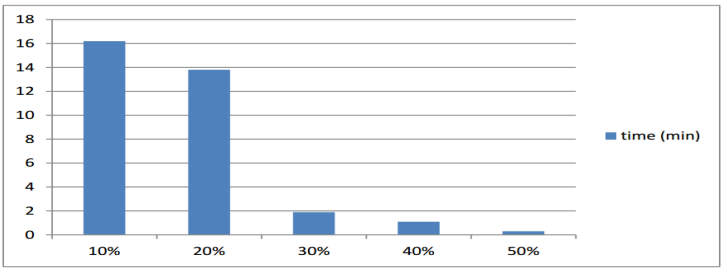


Fig. 5. Computation times for 10 epochs decision strategy

is feasible to compute the decision strategy, but setting a low threshold will take too much time for the recommendation to be generated for real-time software. One can also note that there is a gap between the processing times for 20% and 30% thresholds. This is due to the fact that all the paths of the DDM were explored for a 10% threshold while there were a lot of branches that were not explored for the 30% threshold.

6 Related Work

The work introduced in this paper may seem somewhat similar to providing recommendations in web-based systems. There, too, is a large log with user actions (e.g. what items were purchased) based on which some recommendations are provided. But the fundamental approach focuses on the associations between properties of items. We do not focus on the choice itself but on the reasoning process that leads to that choice. Therefore, the work done in content-based or collaborative filtering poorly fits our needs.

The workflow approaches focus mostly on the control flow perspective[8]. Instead, we try to strike a balance between the data flow perspective aimed at producing a choice and the flow of operations. Considering this focus, we found our inspirationin the Product Based Workflow Modeling approach[6]. The approach has at its core a model called Product Data Model (PDM). The DDM is derived from the PDM, having some specific features and properties[5].

Given a PDM, providing recommendations of the next action to be taken can be done using Markov Decision Process (MDP) approach [2]. In is suggested that the MDP-based recommendation is difficult to apply for real situations because of the State Explosion problem. It was concluded that the state space explodes so it is not feasible to calculate the global optimum path. Instead, it was proven that the strategy of selecting local optimums for each decision epoch yields results close to the global optimum. Even more, also in [2], the objective functions are the overall cost or the overall time of processing. For our approach, the more frequent an operation is performed, the more important it is, so, the objective function of the algorithm is the highest frequency. A DDM-specific property is that an operation can only be executed successfully (since it is a mathematic computation), only once (since it is useless to calculate something once you know its value), and the order of operations for basic data is irrelevant. In [2] these assumptions are ‘forced’ on the PDM to keep computational efforts reasonable.

Schonenberg[8] approached a similar problem of trying to balance between imperative and declarative processes. There is, too, a recommendation algorithm that considers the history of the current trace and aims to provide guidance. It does that based on similar traces while in our approach there is a model that can be exploited.

From artificial intelligence field there are many algorithms that search graphs [9]. We found that A* algorithm also fits our problem and adapted it in [4].

7 Conclusions

The main goal of this paper is to prove that an optimal global decision can be computed, based on an aggregated DDM. Vanderfeesten reported that using the MDP on the more general, design PDM is not feasible due to the need to compute the state-space. Some assumptions and the use of a local decision strategy reduced the state space explosion. However, an aggregated DDM is mined from many decision traces rather than created by an expert. Therefore it has specific properties like the frequency of operations, the fact that each operation is always executed successfully and that the operations producing basic data items can be done in any order. Those properties allowed us to map our recommendation problem successfully to the MDP. Therefore, we are able to argue that it is possible to recommend a global decision strategy that would lead the decision maker through the most important operations in an aggregated DDM.

Considering the aim of this paper, we did not provide a‘real’ validation of the algorithm. In the future papers we will provide comparisons and an evaluation of the algorithm’s results, to prove that it produces the optimal global decision. We will also conduct live controlled experiments to compare the MDP recommendation with the other two recommender systems based on DDMs (using Greedy and A*-like approaches) to determine which ones are preferred in real life situations.

Acknowledgments. This research was supported by Human Resources Development Operational Program through the project Transnational Network of Integrated Postdoctoral Research in the Field of Science Comunication, Capacity Building (Post-doctoral School) and Scholarship Program (CommScie) POSDRU/89/1.5/S/63663.

References

1. Dean, J.W., Sharfman, M.P.: Does Decision Process Matter? A Study of Decision-Making Effectiveness. J. Academy of Management 39, 368–396 (1996)
2. Vanderfeesten, I.T.P., Reijers, H.A., van der Aalst, W.M.P.: Product-based Workflow Support. J. Information Systems 36, 517–535 (2011)
3. Petrusel, R., Vanderfeesten, I., Dolean, C.C., Mican, D.: Making Decision Process Knowledge Explicit Using the Decision Data Model. In: Abramowicz, W. (ed.) BIS 2011. LNBIP, vol. 87, pp. 172–184. Springer, Heidelberg (2011)
4. Petrusel, R., Stanciu, P.L.: Making Recommendations for Decision Processes Based on Aggregated Decision Data Models. In: Abramowicz, W., Kriksciuniene, D., Sakalauskas, V. (eds.) BIS 2012. LNBIP, vol. 117, pp. 272–283. Springer, Heidelberg (2012)

5. Petrusel, R.: Aggregating Individual Models of Decision-Making Processes. In: Ralyté, J., Franch, X., Brinkkemper, S., Wrycza, S. (eds.) CAiSE 2012. LNCS, vol. 7328, pp. 47–63. Springer, Heidelberg (2012)
6. Reijers, H.A., Limam Mansar, S., van der Aalst, W.M.P.: Product-Based Workflow Design. Journal of Management Information Systems 20, 229–262 (2003)
7. Puterman, M.L.: Markov Decision Processes. Discrete Stochastic Dynamic Programming. Wiley, New York (1994)
8. van der Aalst, W.M.P.: Process Mining. Discovery, Conformance and Enhancement of Business Processes. Springer, Heidelberg (2011)
9. Reijers, H.A., Limam, S., van der Aalst, W.M.P.: Product-based Workflow Design. J. of Management Information Systems 20, 229–262 (2003)
10. Schonenberg, H., Weber, B., van Dongen, B.F., van der Aalst, W.M.P.: Supporting Flexible Processes Through Recommendations Based on History. In: Dumas, M., Reichert, M., Shan, M.-C. (eds.) BPM 2008. LNCS, vol. 5240, pp. 51–66. Springer, Heidelberg (2008)
11. Russell, S.J., Norvig, P.: Artificial Intelligence: A Modern Approach. Prentice Hall, Upper Saddle River (2003)

A Literature Survey on Information Logistics*

Bernd Michelberger[1], Ralph-Josef Andris[2], Hasan Girit[1], and Bela Mutschler[1]

[1] University of Applied Sciences Ravensburg-Weingarten, Germany
{bernd.michelberger,hasan.girit,bela.mutschler}@hs-weingarten.de
[2] Center for ERP-Systems, University of Augsburg, Germany
ralph.andris@wiwi.uni-augsburg.de

Abstract. The notion of *information logistics* (IL) has been introduced as a new information management paradigm. Goal is to enable the effective and efficient delivery of needed information in the right format, granularity and quality, at the right place, at the right point in time to the right actors. IL has received much attention in recent years, both from researchers and practitioners. In order to better understand the state-of-the-art and current research trends in the research field of IL, this paper presents a comprehensive IL literature survey. In total, we identified 53 scientific articles discussing IL concepts and approaches. These articles were systematically analyzed and finally classified in ten research clusters. Based on these clusters, a more comprehensive understanding of past, current, and future IL developments becomes possible.

Keywords: information logistics, literature survey.

1 Introduction

Today's information and communication technologies (ICT) enable the access to information from any location and at any time. At the same time, users are confronted with a continuously increasing information overload [1] making it difficult for them to identify, handle, and apply information.

In order to cope with this challenge, the idea of *information logistics* (IL) has been introduced. Goal is to enable the effective and efficient delivery of needed information in the right format, granularity and quality, at the right place, at the right point in time to the right actors. To achieve this goal, basic principles from many research areas such as material logistics and lean management have been both adopted and adapted. Generally, IL is independent on the use of ICT, but ICT, of course, can be seen as an IL-enabler [2].

In this paper, we present a comprehensive literature survey on the state-of-the-art in the research field of IL. The main objective of our survey is to better understand past, current, and future developments in IL. More precisely, our research questions are: What is the state-of-the-art and what are current research

* This paper was done in the niPRO research project. The project is funded by the German Federal Ministry of Education and Research (BMBF) under grant number 17102X10. More information can be found at http://www.nipro-project.org.

W. Abramowicz (Ed.): BIS 2013, LNBIP 157, pp. 138–150, 2013.

trends in the research field of IL? To answer these questions, we analyzed 53 IL-related articles and classified them in ten research clusters.

The remainder of this paper is organized as follows. Section 2 describes the research methodology underlying our survey. Section 3 presents main results of the survey. Section 4 discusses our results. Section 5 summarizes related work and Section 6 concludes the paper with a summary.

2 Research Methodology

In order to ensure the validity of our literature survey, we used survey protocol documents as proposed in the literature survey guide by Okoli and Schabram [3]. Our survey comprises four consecutive steps (cf. Fig. 1): (1) search, (2) selection, (3) analysis, and (4) classification.

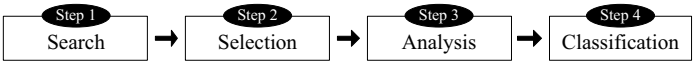


Fig. 1. Steps of our literature survey

Step 1: First, a profound web-based search was conducted to identify potentially relevant IL articles. We considered an article as being relevant based upon two selection criteria: (1) an article contains the term "information logistics" in its title and (2) the article has to be written in English. Specifically, we used Google Scholar, SpringerLink, the Association for Computing Machinery (ACM) Digital Library, the Institute of Electrical and Electronics Engineers (IEEE) Xplore Digital Library, ScienceDirect, and Microsoft Academic Search (AS). We considered articles from books, journals, and both conference and workshop proceedings. We also took into account reports, editorials, and PhD theses. Other kinds of articles such as commercial white papers were not considered.

Step 2: In the second step, we reassessed the number of articles identified in Step 1. In particular, we removed both irrelevant articles (e.g., an article with the title "Information, Logistics and Retailing Services") and duplicate ones (of course, some articles have been found by several search engines). Then, we identified and selected analyzable articles. We considered an article as analyzable if the article's full text was available. Finally, we enriched all remaining articles with metadata such as citation count, type of publication, and year of publication. This allowed for a more in-depth analysis (cf. Step 3) and also supported the subsequent clustering of the articles (cf. Step 4). In total, we had a list of 63 relevant articles potentially being relevant at the end of Step 2.

Step 3: In the third step, we performed an in-depth content analysis of the 63 articles. Therefore, all 63 articles were reviewed by at least two researchers according to the procedures suggested in [3]. Among other things, a separate review was created for each article. Note that based on the reviews ten articles

were excluded from the survey due to quality issues or other reasons. For example, some articles did not meet our content requirements, consisted only of a few sentences or were literature surveys similar to our one.

Step 4: Based on the remaining 53 articles, the generated meta data, and the created reviews, we then performed the clustering in the last step. Thereby, for example, we also took into account topic, author and institutional relationships. Finally, we organized 53 articles in ten research clusters.

Note that our literature survey has several limitations. First, we only considered articles with "information logistics" in their title. This limitation was made due to the large amount of search engine hits we obtained when we considered papers with the term "information logistics" in their full text. Second, only articles in English were considered.

3 The Survey

This section summarizes the main results of our survey. Section 3.1 discusses the data collection for our literature survey. Section 3.2 presents the ten identified IL research clusters (C1 to C10).

3.1 Data Collection

Altogether, our initial web-based search resulted in 282 hits, i.e., 282 articles potentially being relevant for our survey. Google Scholar delivered the most hits (139 hits), followed by Microsoft AS (94 hits), and the IEEE Xplore Digital Library (20 hits). Less results have been identified based on the ACM Digital Library (13 hits), SpringerLink (13 hits), and ScienceDirect (3 hits). Table 1 summarizes the raw results collected during Step 1.

In Step 2, we identified articles which did not meet our selection criteria (cf. Section 2). As a result of this, we excluded 125 articles from the study, i.e., 157

Table 1. Raw results

	total hits (Step 1)	**irrelevant hits** (Step 2)	**relevant hits** (Step 2)
Google Scholar	139	62	77
SpringerLink	13	1	12
ACM Library	13	0	13
IEEE Library	20	9	11
ScienceDirect	3	0	3
Microsoft AS	94	53	41
total hits	**282**	**125**	**157**

articles remained, implying an aggregated precision across all search engines of 55.67 %. Out of these 157 articles we then removed duplicate articles and also excluded articles we could not analyze due to a missing full text. At the end of this step, 63 articles were selected for further in-depth analysis.

Before starting our analysis (i.e., Step 3), each of the 63 articles was assigned with additional metadata (cf. Section 2). Among other things, the year of publication was documented. This enabled us, for example, to look for time-based trends and developments. Figure 2 shows, for example, that the number of IL-related articles has significantly increased in recent years.

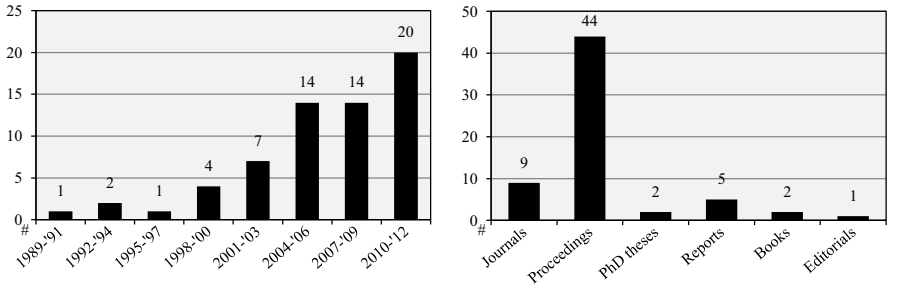


Fig. 2. Publication date and type of analyzed articles

Figure 2 also illustrates the type of the considered IL articles. Most IL articles (44 ones) stem from workshop or conference proceedings, followed by journals (9 articles), reports (5 articles), PhD theses (2 articles), articles in books (2 articles), and editorials (1 article).

Figure 3 illustrates the citation count of the articles. Most articles (21 ones) are not cited. 14 articles have 1-2 citations, 10 articles have 11-20 citations, 7 articles have 3-5 citations, and another 7 articles have 6-10 citations. The most three cited articles are [4], [5], and [6] (according to Google Scholar).

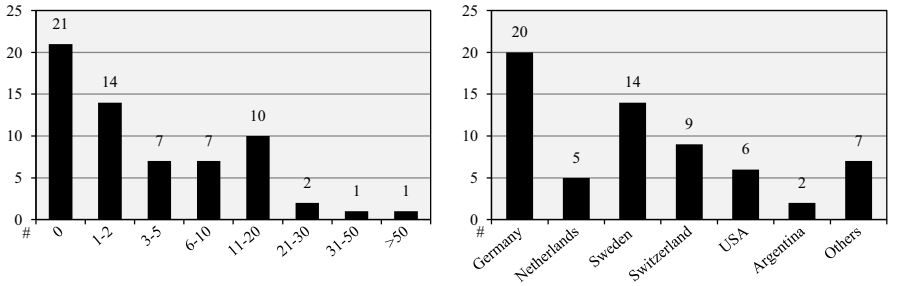


Fig. 3. Citation counts and countries of origin

Figure 3 also illustrates the country of origin of the articles. Most articles (20 ones) stem from Germany, followed by Sweden (14 articles), Switzerland (9 articles), USA (6 articles), and The Netherlands (5 articles).

In Step 3, the 63 articles were carefully reviewed by at least two reviewers. For each article, a review containing a short summary, the full abstract, and key words was created. As aforementioned, we excluded ten further articles from the survey as a result of the reviews due to quality issues or other reasons. Thus, 53 articles were finally included in the literature survey.

3.2 Research Clusters

This section describes the ten IL research clusters (cf. Fig. 4) we identified based on our literature survey. Table 2 additionally shows the most cited paper for each cluster. Table 3 summarizes the main characteristics of each cluster.

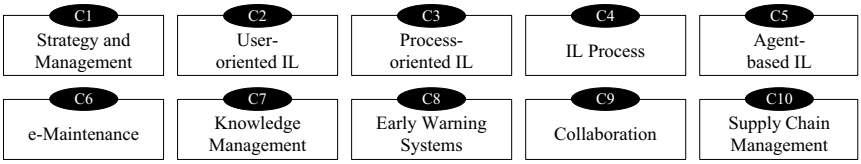


Fig. 4. Identified research clusters.

Cluster 1 (C1): Strategy and Management. Most articles from this cluster stem from the Management Institute of the University of St. Gallen (Switzerland). All articles belonging to this cluster concern strategy and management issues related to IL, in particular the transformation of enterprises into IL organizations. [7], for example, discusses the state of IL strategy. The main finding is that IL strategy depends on company size and structure. In addition, [8] investigates critical success factors for IL strategies. Examples of identified success factors include comprehensiveness, flexibility, support, communication, IT strategy orientation, business/IT partnership, and project collaboration. Special focus of [9] are IL management tasks enabling the use of IL concepts within an organization. [10] discusses general and thus very broad IL management challenges. More specific conceptual models to better understand IL requirements in enterprises are presented in [11] and [12]. A case study assessing the IL landscape of a global automotive company is presented in [13]. Another empirical study assessing benefits, design factors, and realization approaches in IL is presented in [5]. Finally, [14] presents a case study on the design and implementation of IL in the healthcare domain.

Cluster 2 (C2): User-oriented IL. The articles in this cluster address the challenges in user-oriented IL. In [15], the author discusses challenges and solutions for user-oriented information supply in IL. According to [6], IL can be understood as an approach enabling just-in-time delivery of information to users. Corresponding examples are given in the fields of wearable computing

[16], weather forecast [17], and the healthcare domain [18]. [19] argues that the success of information supply depends on successful user adoption and powerful frontend technologies. Therefore, in [19], a Twitter-like frontend for IL is presented. Moreover, [20] presents intelligent IL services and also discusses integration challenges. In [21], an industrial case study on these IL services is presented. A similar, but more technical perspective on integration challenges in IL is addressed in [22].

Besides, context-awareness adopts a key role in user-oriented IL. [23], for example, presents a study on context-based models for IL. Context definitions and representations from different viewpoints (e.g., information demand analysis, decision support) are presented [24]. A reference architecture for context-awareness in IL applications is presented in [2]. Another context framework for IL also considering various situation) is presented by [4]. This framework has been tested in [25] using an automotive prototype to demonstrate its general applicability.

Cluster 3 (C3): Process-Oriented IL. This cluster deals with the alignment of process-related information (e.g., working instructions, best practices etc.) with knowledge-intensive business processes so that decision-makers and knowledge-workers can perform their tasks in the best possible way [26]. Specifically, process-oriented IL enables process-oriented and context-aware delivery of relevant information to knowledge-workers. For this task a semantic information network is used, which integrates process objects, information objects, as well as their relationships. In [27], quality dimensions of process-related information (e.g., completeness, punctuality etc.) are investigated in order to determine the relevance of information along business processes. In [28], an ontology-based context framework for process-oriented IL is proposed. This framework aims at the context-aware delivery of process-related information to process participants.

Cluster 4 (C4): IL Process. This cluster is mainly addressed by the Jönköping Business School in Sweden. In [29], IL is introduced as an approach (or process) transforming a given input (e.g., a project description, lessons learned) into some form of output (e.g., a best practice document). Goal is to transform fragmented information into usable information for the receiver. An IL transformation comprises three phases: information supply, information production, and information distribution. In order to realize this IL approach, [30] suggests an agent-based IL approach (i.e., the combination of multi-agent systems and IL). In [31], the notion of IL and basic ingredients of the IL process are discussed. Finally, in [32], the authors present a visual knowledge modeling approach of an IL process as defined in [31].

Cluster 5 (C5): Agent-Based IL. This cluster concerns agent-based IL. In this context, an agent is a piece of software that acts for a user when searching for needed information. [33], for example, argues that a multi-agent IL approach, providing techniques for autonomous, situated, social, and pro-active information services, is a well-suited approach for realizing IL. A different perspective is adopted in [34]. The authors discuss the use of adaptive multi-agents approaches. [35] presents an agent-based IL architecture for process management, i.e., to support processes which rely on informational inputs and produce information as

an output. Finally, [36] presents an agent-based IL approach for monitoring and coordination of processes.

Cluster 6 (C6): e-Maintenance. The articles in this cluster concern IL in the context of e-Maintenance. One central maintenance problem is to manage system complexity. Some experiences from the aerospace domain are described in [37]. Specific e-Maintenance IL solutions are discussed in [38]. Moreover, [39] proposes a framework for IL-driven e-Maintenance. In [40], maintenance and ICT are merged from an IL perspective. The role of IL and data warehousing in maintenance management is addressed in [41].

Cluster 7 (C7): Knowledge Management. The articles in this cluster deal with knowledge processing in and through IL. [42] and [43], for example, discuss an IL approach for knowledge processing. The presented knowledge processing approach aims at increasing the daily performance of knowledge-workers in enterprises. [44] proposes IL for conceptual correspondence monitoring. Finally, [45] and [46] address the enabling role of IL approaches in knowledge management. They conclude that an IL approach significantly improves a knowledge-worker's daily performance.

Cluster 8 (C8): Early Warning Systems. This cluster is mainly addressed by the German Research Centre for Geosciences. [47] and [48] apply the concept of IL to hazard monitoring and early warning systems. Goal is to enable the generation of user-tailored warning messages considering user needs with respect to content, location, or individual requirements. In addition, filter mechanisms to avoid information overload in emergency situations are presented.

Cluster 9 (C9): Collaboration. This cluster discusses the importance of IL to support collaboration in enterprises. In [49] and [50], IL is defined as the maintenance, tracking, monitor, and enactment of information flows within collaborative environments. [51] argues, in addition, that an IL approach is necessary to cope with the complexity of information flows. [52] analyzes the information flow between participants of collaborative processes.

Cluster 10 (C10): Supply Chain Management. This cluster deals with IL approaches supporting Supply Chain Management. [53], for example, proposes the design of an ontology to support IL supply chains. This ontology is described in more detail in [54]. Besides, [55] proposes a supply chain strategy to increase supply chain integration through organizational learning regarding IL activities.

Table 2. Most cited article in each cluster

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
Article	[5]	[4]	[28]	[30]	[34]	[37]	[46]	[47]	[50]	[54]
Date of Article	2008	2004	2012	2008	2001	2009	2009	2011	2000	2005
Citation Count	27	56	3	11	13	25	6	7	12	1
Type	Proc.	Proc.	Proc.	Proc.	Proc.	Jour.	Repo.	Proc.	Proc.	Jour.

Table 3. Articles in the research clusters

	C1	**C2**	**C3**	**C4**	**C5**	**C6**	**C7**	**C8**	**C9**	**C10**
Date of First Article	1993	1999	2011	2003	2000	2009	2008	2011	2000	2005
Date of Latest Article	2012	2012	2012	2008	2007	2010	2011	2012	2004	2006
Trend in Cluster	↗	↗	↗	↓	↓	→	↗	→	↓	↓
Foundation in Cluster	↑	↑	↘	↗	↗	↑	→	→	↗	↘
Articles in Cluster	↗	↑	→	→	→	↗	↗	↘	→	→
1989-'91	-	-	-	-	-	-	-	-	-	-
1992-'94	1	-	-	-	-	-	-	-	-	-
1995-'97	-	-	-	-	-	-	-	-	-	-
1998-'00	-	1	-	-	1	-	-	-	2	-
2001-'03	-	4	-	1	1	-	-	-	1	-
2004-'06	-	5	-	1	1	-	-	-	1	3
2007-'09	5	1	-	2	1	3	2	-	-	-
2010-'12	3	3	3	-	-	2	3	2	-	-
total	**9**	**14**	**3**	**4**	**4**	**5**	**5**	**2**	**4**	**3**

4 Discussion

The number of IL-related articles, both from researchers and practitioners, has significantly increased in recent years. Consider, for example, the last three years: 20 new articles have been published since 2010. This makes it worthwhile to conduct a survey. As can be seen, we were able to identify a large number of IL methods, concepts, and approaches for our literature survey. The main problem: The broad field of IL makes the comparison of methods, concepts, and approaches a challenge. In fact, the term "information logistics" is the only commonality between many IL articles [56].

Reason is that IL addresses and recombines a large number of well-known research areas, e.g., material logistics [6], process management [26], information management [9], ubiquitous computing [2], or semantic technologies [15]. Additionally, ideas from business intelligence, location-based services, or enterprise content management are picked up as well.

We classified articles along ten research clusters in our study. However, there do exist overlaps between these clusters (also meaning that several of the identified IL articles could be assigned to more than one cluster). For example, both C2 (i.e., user-oriented IL) and C3 (i.e., process-oriented IL) focus on the delivery of needed information to users. However, while C2 concerns respective requirements and solutions for human users [6], C3 focuses on the support of both

business processes and process participants (as articles assigned to C2 neglect business processes and process orientation). Still, topics are similar in C2 and C3. As another example for overlapping clusters consider C4 (i.e., IL processes) and C5 (i.e., agent-based IL). In order to establish IL processes, [30] (assigned to C4) suggests to use an agent-based IL approach, like the one introduced in [36] (assigned to C5). Also consider C3 and C5. In [35], an agent-based IL architecture for process management is given. This work, however, could be also assigned to C3. In addition, C7 (i.e., knowledge management) and C10 (i.e., supply chain management) do also overlap. For example, both [42] (from C7) and [53] (from C10) discuss ontologies in the context of IL. Finally, IL-based early warning systems [48] in C8 (i.e., early warning systems) adopt approaches we assigned to C2 (e.g., the weather forecast prototype [17]).

5 Related Work

There already exist surveys dealing with IL. However, these surveys either address specific IL application domains or do only include articles published until 2009. More specifically, Haftor [57] conducts a first study on IL definitions and proposes a novel notion of IL. Similar to our survey (cf. Table 4), in turn, is the second study conducted by Haftor et al. [56]. However, this survey does only include IL articles which have been published until 2009. As there have been many IL publications since 2009, our survey represents the most current study. In addition, unlike the study of Haftor et al. [56], we discuss overlaps between research clusters and also discuss time-based trends in IL, types of articles, number of citations, and the country of origin of articles.

Table 4. Differences between [56] and our literature survey

	Haftor et al. [56]	**Our literature survey**
Period investigated	until 2009	until 2012
Languages of Articles	English, German, Swedish	English
Number of Articles	71	63
Number of Articles in English	~ 35	63
Strengths of Clusters	■	□
Limitations of Clusters	■	□
Time-based Trends in IL	□	■
Types of Articles	□	■
Citation Counts of Articles	□	■
Country of Origin of Articles	□	■

□ = no ■ = yes

6 Summary

This paper summarizes the results of a profound literature survey in the field of IL. The main objective of our survey is to better understand past, current, and future developments in IL. In total, we included 53 articles in the survey. These 53 articles have been classified into ten research clusters.

References

1. Öhgren, A., Sandkuhl, K.: Information Overload in Industrial Enterprises - Results of an Empirical Investigation. In: Proc. 2nd ECIME 2008, pp. 343–350 (2008)
2. Haseloff, S.: Context Awareness in Information Logistics. PhD Thesis, Technical University of Berlin (2005)
3. Okoli, C., Schabram, K.: A Guide to Conducting a Systematic Literature Review of Information Systems Research. Sprouts: Working Papers on Information Systems 10(26) (2010)
4. Meissen, U., Pfennigschmidt, S., Voisard, A., Wahnfried, T.: Context- and Situation-Awareness in Information Logistics. In: Lindner, W., Fischer, F., Türker, C., Tzitzikas, Y., Vakali, A.I. (eds.) EDBT 2004. LNCS, vol. 3268, pp. 335–344. Springer, Heidelberg (2004)
5. Bucher, T., Dinter, B.: Process Orientation of Information Logistics - An Empirical Analysis to Assess Benefits, Design Factors, and Realization Approaches. In: Proc. 41st HICSS 2008, pp. 392–402 (2008)
6. Deiters, W., Löffeler, T., Pfennigschmidt, S.: The Information Logistics Approach: Toward User Demand-driven Information Supply. In: Proc. CMSD 2003, pp. 37–48 (2003)
7. Dinter, B., Winter, R.: Information Logistics Strategy - Analysis of Current Practices and Proposal of a Framework. In: Proc. 42nd HICSS 2009, pp. 1–10 (2009)
8. Dinter, B.: Success Factors for Information Logistics Strategy - An Empirical Investigation. J. of DSS (2012)
9. Klein, S.: Information Logistics. J. of EM 3(3), 11–12 (1993)
10. Winter, R.: Enterprise-wide Information Logistics: Conceptual Foundations, Technology Enablers, and Management Challenges. In: Proc. 30th ITI 2008, pp. 41–50 (2008)
11. Lahrmann, G.: Strategic Positioning of Information Logistics Service Providers: Guidelines for Selecting Appropriate Organizational Models. In: Proc. 43rd HICSS 2010, pp. 1–10 (2010)
12. Lahrmann, G., Stroh, F.: Towards a Classification of Information Logistics Scenarios - An Exploratory Analysis. In: Proc. 42nd HICSS 2009 (2009)
13. Winter, R., Bischoff, S., Wortmann, F.: Revolution or Evolution? Reflections on in-memory Appliances from an Enterprise Information Logistics Perspective. In: Proc. IMDM 2011, pp. 23–34 (2011)
14. Bucher, T., Teich, J.M.: Design and Implementation of a Computerized Physician Order Entry System at Brigham and Women's Hospital - A Case Study on Process Orientation of Information Logistics. Working Paper, St. Gallen (2008)
15. Sandkuhl, K.: Information Logistics in Networked Organizations: Selected Concepts and Applications. In: Filipe, J., Cordeiro, J., Cardoso, J. (eds.) ICEIS 2007. LNBIP, vol. 12, pp. 43–54. Springer, Heidelberg (2008)

16. Heuwinkel, K., Deiters, W., Königmann, T., Löffeler, T.: Information Logistics and Wearable Computing. In: Proc. 23rd Workshops ICDCS 2003, pp. 283–289 (2003)
17. Jaksch, S., Pfennigschmidt, S., Sandkuhl, K., Thiel, C.: Information Logistic Applications for Information-on-Demand Scenarios: Concepts and Experiences from WIND Project. In: Proc. EUROMICRO 2003, pp. 141–147 (2003)
18. Heuwinkel, K., Deiters, W.: Information Logistics, e-Healthcare and Trust. In: Proc. IADIS 2003, vol. 2, pp. 791–794 (2003)
19. Böhringer, M., Gaedke, M.: Ubiquitous Microblogging: A Flow-Based Front-End for Information Logistics. In: Abramowicz, W., Tolksdorf, R., Węcel, K. (eds.) BIS 2010. LNBIP, vol. 57, pp. 158–167. Springer, Heidelberg (2010)
20. Sandkuhl, K., Smirnov, A., Shilov, N.: Information Logistics in Engineering Change Management: Integrating Demand Patterns and Recommendation Systems. In: Niedrite, L., Strazdina, R., Wangler, B. (eds.) BIR Workshops 2011. LNBIP, vol. 106, pp. 14–25. Springer, Heidelberg (2012)
21. Sandkuhl, K., Borchardt, U., Lantow, B., Stamer, D., Wißotzki, M.: Towards Adaptive Business Models for Intelligent Information Logistics in Transportation. In: Proc. 11th BIR 2012, pp. 18–30 (2012)
22. Vermeer, B.H.P.J.: Information Logistics: A Data Integration Method for Solving Data Quality Problems with Article Information in Large Interorganizational Networks. In: Proc. ICIQ 1999 (1999)
23. Levashova, T., Lundqvist, M., Sandkuhl, K., Smirnov, A.: Context-based Modelling of Information Demand: Approaches from Information Logistics and Decision Support. In: Proc.14th ECIS 2006, pp. 1511–1522 (2006)
24. Lundqvist, M., Sandkuhl, K., Levashova, T., Smirnov, A.: Context-Driven Information Demand Analysis in Information Logistics and Decision Support Practices. In: Proc. 1st Workshop CO: TPA, pp. 124–127 (2005)
25. Meissen, U., Pfennigschmidt, S., Wahnfried, T., Sandkuhl, K.: Situation-based Message Rating in Information Logistics and its Applicability in Collaboration Scenarios. In: Proc. 30th EUROMICRO 2004, pp. 484–491 (2004)
26. Michelberger, B., Mutschler, B., Reichert, M.: Process-oriented Information Logistics: Aligning Enterprise Information with Business Processes. In: Proc. 16th EDOC 2012, pp. 21–30 (2012)
27. Michelberger, B., Mutschler, B., Reichert, M.: Towards Process-oriented Information Logistics: Why Quality Dimensions of Process Information Matter. In: Proc. 4th EMISA 2011, pp. 107–120 (2011)
28. Michelberger, B., Mutschler, B., Reichert, M.: A Context Framework for Process-oriented Information Logistics. In: Abramowicz, W., Kriksciuniene, D., Sakalauskas, V. (eds.) BIS 2012. LNBIP, vol. 117, pp. 260–271. Springer, Heidelberg (2012)
29. Apelkrans, M., Håkansson, A.: Enterprise Systems Configuration as an Information Logistics Process - A Study. In: Proc. 9th ICEIS 2007, pp. 212–220 (2007)
30. Apelkrans, M., Håkansson, A.: Information Coordination Using Meta-agents in Information Logistics Processes. In: Lovrek, I., Howlett, R.J., Jain, L.C. (eds.) KES 2008, Part III. LNCS (LNAI), vol. 5179, pp. 788–798. Springer, Heidelberg (2008)
31. Apelkrans, M.: Multi Project Control As An Information Logistics Process Topic Area: Knowledge Management. In: Proc. AIPM 2003 (2003)
32. Apelkrans, A., Håkansson, A.: Visual Knowledge Modeling of an Information Logistics Process - A Case Study. In: Proc. ICICKM 2005 (2005)

33. Knublauch, H., Rose, T.: Application Scenarios of Agent-Based Information Logistics in Clinical and Engineering Domains. In: Proc.14th ECAI 2000, pp. 85–88 (2000)
34. Timm, I.J., Woelk, P.O., Knirsch, P., Tönshoff, H.K., Herzog, O.: Flexible Mass Customisation: Managing its Information Logistics Using Adaptive Co-operative Multiagent Systems. In: Proc. 6th ISL, pp. 227–232 (2001)
35. Winkler, S., Zimmermann, R., Bodendorf, F.: An Agent-Based Information Logistics Architecture for Process Management. In: Proc. CIMCA 2005, pp. 745–751 (2005)
36. Bodendorf, F., Winkler, S., Zimmermann, R., Vögele, B.: Agent-Based Information Logistics in Planning Processes. In: Proc. 2nd ICONS 2007 (2007)
37. Candell, O., Karim, R., Sölderholm, P.: eMaintenance-Information Logistics for Maintenance Support. J. of RaCIM 25(6), 937–944 (2009)
38. Candell, O., Karim, R., Söderholm, P., Kumar, U.: Service-oriented Information Logistics as Support to Intelligent Transport Services. In: Proc. 16th ITS 2009, pp. 21–25 (2010)
39. Haftor, D.M., Kajtazi, M., Mirijamdotter, A.: Research and Practice Agenda of Industrial e-Maintenance: Information Logistics as a Driver for Development. In: Proc. 1st Workshop and Congress on eMaintenance (2010)
40. Karim, R., Candell, O., Sölderholm, P.: E-Maintenance and Information Logistics: Aspects of Content Format. J. of QME 15(3), 308–324 (2009)
41. Vieira, A.C.V., Cardoso, A.J.M.: The Role of Information Logistics and Data Warehousing in Educational Facilities Asset Management. J. of Syst. Assur. Eng. Manag. 1(3), 229–238 (2010)
42. Czejdo, B.D., Baszun, M.: Information Logistics for Incomplete Knowledge Processing. In: Proc. CCIS 2010, pp. 295–302 (2010)
43. Czejdo, B.D., Baszun, M., Cummings, T.: Knowledge Workers' Advisor Based On Information Logistics Models. In: Proc. 4th KGCM 2010 (2010)
44. Rudzajs, P., Kirikova, M.: Multimode Information Logistics for Conceptual Correspondence Monitoring. In: Proc. 11th BIR 2012, pp. 31–42 (2012)
45. Willems, J.: From Having to Using: Information Logistics Experience Centre is Born. Working Paper, Nyenrode (2008)
46. Willems, A., Willems, J., Hajdasinski, A.: Information Logistics Research Report - Frameworks in the Healthcare Industry. Working Paper, Nyenrode (2009)
47. Lendholt, M., Hammitzsch, M.: Generic Information Logistics for Early Warning Systems. In: Proc. 8th ISCRAM 2011 (2011)
48. Lendholt, M., Hammitzsch, M.: Towards an Integrated Information Logistics for Multi Hazard Early Warning Systems. J. of OEE, 27–42 (2012)
49. Soibelman, L., Caldas, C.: Information Logistics Approach for Construction Inter-Organizational Information Systems. In: Proc. ICIT 2000 and CIB W078 (2000)
50. Soibelman, L., Caldas, C.: Information Logistics for Construction Design Team Collaboration. In: Proc. 8th ICCCBE-VIII 2000 (2000)
51. Scherer, R.J.: Information Logistics for Supporting the Collaborative Design Process. In: Bento, J., Duarte, J.P., Heitor, M.V., Mitchell, W.J. (eds.) Collaborative Design and Learning: Competence Building for Innovation, pp. 198–222 (2004)
52. Nuntasunti, S.: The Effects of Visual-based Information Logistics in Construction. PhD Thesis, North Carolina (2003)
53. Vegetti, M., Gonnet, S., Henning, G., Leone, H.: Information Logistics for Supply Chain Management within Process Industry Environments. J. of CACE 20, 1231–1236 (2005)

54. Vegetti, M., Gonnet, S., Henning, G., Leone, H.: Towards a Supply Chain Ontology of Information Logistics within Process Industry Environments. In: Proc. ENPROMER 2005 (2005)
55. Timlon, J., Harryson, S.: Realizing a New Supply Chain Strategy – Re-Conceptualizing Actors: Meaning Structures of Information Logistics Activities. In: Annual IMP Conference, Milan (2006)
56. Haftor, D.M., Kajtazi, M., Mirijamdotter, A.: A Review of Information Logistics Research Publications. In: Proc. 3rd Workshop ILOG 2010, pp. 244–255 (2011)
57. Haftor, D.: Information Logistics: A Proposed Notion. In: Proc. 11th BIR 2012, pp. 60–78 (2012)

Knowledge Discovery Methods for Bankruptcy Prediction

František Babič, Cecília Havrilová, and Ján Paralič

Department of Cybernetics and Artificial Intelligence,
Faculty of Electrical Engineering and Informatics,
Technical university of Košice, Letná 9/B, Košice, 042 01 Košice, Slovakia
{frantisek.babic,cecilia.havrilova,jan.paralic}@tuke.sk

Abstract. Business bankruptcy is a negative phenomenon, whose symptoms can be identified in advance by means of financial data analyses. The aim of this paper is to present two experimental studies using two different approaches to analyze company's financial situation based on selected financial indicators. The first approach used data from financial database called Amadeus to generate a binary prediction model to evaluate a possible future financial health status of the EU companies using suitable machine learning algorithms. The second one included a design and creation of data warehouse based on data from two financial databases Albertina and Creditinfo (SK and CZ companies) to evaluate financial health status of the companies from Slovakia and Czech Republic through index of bankruptcy IN_{99}.

Keywords: bankruptcy, decision trees, k-nearest neighbor, data warehouse.

1 Introduction

Effective and simple understanding of companies' financial situation represents a significant source of information for decision activities on the management level. Hidden knowledge, patterns or relations can be extracted from collected corporate data by several approaches: from simple statistical methods implemented in MS Excel with visualization in the form of different graphs; through creation of central data warehouses aggregating all heterogeneous data into one place for further OLAP (online analytical processing) analysis; up to design and implementation of data mining streams based on available machine learning methods. Common aim of all these approaches is to obtain accurate important information on time to efficiently support the decision processes.

Bankruptcy represents a legal status of a person or a company that cannot repay the debts they owe to the creditors. Early identification of typical bankruptcy indicators provides important information for the creditors or investors in evaluating the possibility that a company may go bankrupt. This is a complex process with many relevant inputs and dependencies. One possibility to support decision makers in this process is to use knowledge discovery methods. Discovered predictive models need to be properly evaluated on different datasets in order to cover as many different situations (reasons of bankruptcy) as possible.

W. Abramowicz (Ed.): BIS 2013, LNBIP 157, pp. 151–162, 2013.

The Business Intelligence (BI) first appeared in 1989 as a common set of concepts and methods for improvements of the decision processes in companies through data analyses, reporting and query tools. The aim is to create a data warehouse containing all preprocessed historical data for analytical purposes. Typical basic architecture of BI solution is composed of the following layers: layer for extraction, transformation and data cleaning, represented by ETL (extract, transform and load)and EAI (enterprise application integration) systems; a data storage (archives) layer containing data warehouses, data marts, operative data storages and temporary data storages; a data analysis layer including OLAP systems, data mining and reporting tools; and a presentation layer consists of different portal solutions, EIS (executive information systems)or other analytical applications.

The paper was motivated by our previous experiences in using knowledge discovery methods in other domains (meteorological data, transactional market data, various logs, etc.) and availability of interesting data source with financial data. This combination provides also the opportunity to test suitability of some supporting software applications for similar tasks. The whole paper starts with short introduction of specified tasks and theoretical basics; then continues with briefly presentation of related works. The next two main sections describe in details both used approaches to analyse financial data of real companies and results that we achieved. The last one summarises the main contributions and inspirations for our future work.

1.1 State of the Art

The aim of this section is to briefly present some existing approaches on how to analyze financial data with respect to bankruptcy prediction. One of the most widely used methods for bankruptcy prediction is neural networks (NN) in various forms. The back propagation network [3] was used in [19] to analyze data from Texas bank (one year and two years before the failure)and the used neural network provided better predictive accuracy that other tested methods (k-nearest neighbors or ID3). Belgian bankruptcy data of 182 banks was used in [18] to generate a prediction models based on extension of typical back propagation algorithm with feature construction methods. Predictive capacity of neural networks and multivariate discriminant analysis is described in [20]. Influence of the selected financial indicators to bankruptcy prediction (one year before) based on back propagation network is presented in [2].

The second interesting direction deals with mathematic-statistical methods that have a wide range of applications, but the prediction process is in this case more demanding. These methods are characterized by their independence from subjective opinions of the experts. Typical representative of the multi-dimensional analyses [10] is Altman classification model [1] incorporating comprehensive inputs. This model is based on seven financial indicators as return of assets, stability of earnings, liquidity, etc. The obtained classification results of this method on the 111 companies ranged from 96% for one year before bankruptcy to 70% for five years.

Exploitation of logistic regression [12] for failure prediction is described in [14], [16] and [8]. The Ohlson's logit model was compared with Case-based reasoning in

[4]. The obtained results were very similar and plausible. Next approach to solve this task includes Formal Concept Analyses (FCA) [9] with its practical implementation through generalized one-sided concept lattices [5] (theoretically based on [17]) that allow exploitation of FCA method on data with different types of attributes [6].The analysis of multiple data tables using FCA method is described in [7].

In the case of BI, existing approaches havenot been widely used so far, so its relevancy for prediction is somewhat disputable. However, some initial studies and experiments are described in [11] and [13].

This survey resulted in some inspiration of our experiments that are described in the following sections. The following section describes a traditional process of data analysis making use of well-known CRISP-DM methodology and selected machine learning algorithms. Section 3 presents a potential of relatively new approach to provide financial analysis in a more simple way without required expert knowledge from knowledge discovery domain. Conclusions and ideas for future work are mentioned in the last section.

2 Bankruptcy Prediction

The first task deals with effective prediction of the bankruptcy through suitable knowledge discovery methods based on available financial data. Created models offer important information about actual financial health of the company in the domain of potential failure. Obtained results can be used to support decision processes of the company managers, investors, business partners or banks. Important input factors of the whole knowledge discovery solution include quality of source data and selection of crucial indicators for mining algorithms.

The main objective is to classify a set of investigated companies into two main categories: bankrupt or active. For this purpose source dataset was extracted from database Amadeus[1] including financial information about more than 15 million public and private companies from 42 European countries. Database is constructed with data from more than 30 specialized regional providers. The resulting models were compared based on their accuracy that was calculated using traditional confusion matrix (see Table 1):

Classification accuracy = (TP + TN) / (TP + TN + FP + FN)

Table 1. Common confusion matrix for evaluation of the created classification models

		Actual class	
Predicted class		Bankrupt (B)	Active (A)
	B	TP (true positive)	FP (false positive)
	A	FN (false negative)	TN (true negative)

[1] https://amadeus.bvdinfo.com

CRISP-DM represents a traditional methodology for complex knowledge discovery processes. The whole process consists of the six main phases with both-sided interactions between them. We used this methodology to create a basic structure for our experiments and to ensure their effective realization with the supporting application SPSS Clementine v10.1[2].

2.1 Data Description

Amadeus database contains financial data from the last 10 years including 26 items from the balance sheet and 32 financial indicators. These indicators can be used for classification purposes, but it is necessary to say that company's financial situation depends of several other factors e.g. political situation or global crises. We started the whole discovery process with initial understanding and evaluation of data stored in the Amadeus database; especially we investigated a level of missing values for respective financial indicators. Based on this evaluation we extracted a representative data sample (basic dataset) containing the records of 16 thousands EU companies from 2003 to 2007 describing by a basic set of 8 attributes like company name, address, country, and a set of 24 indicators, e.g. returns, cash flow, asset turnover, liquidity, length of loan, own capital, ratio of general capital liability, company state, etc. This initial dataset was further modified based on selected processing methods or mining algorithms.

2.2 Data Preparation

At first the basic dataset was divided into four samples following a traditional categorization of the companies or enterprises (Amadeus internal filter):

- Very large companies (returns more than 100 million €, assets more than 200 million €, number of employees more than 1 000)
- Large companies (returns more than 10 million €, assets more than 20 million €, number of employees more than 150)
- Medium companies (returns more than 1 million €, assets more than 2 million €, number of employees more than 15)
- Small companies

This division was motivated by a fact that a size of company represents an important factor with strong influence to financial health and performance of the company. In the case of very large companies we have identified only 33 records for bankruptcy in both sets so this category was eliminated from further experiments.

In the next step the financial indicators with a relatively high number of missing values were eliminated, e.g. average costs per employee, company value, operating incomes, and costs for R&D, working capital per employee, etc. This operation resulted in 20 financial indicators that were further investigated to identify indicators

[2] http://en.wikipedia.org/wiki/SPSS_Modeler

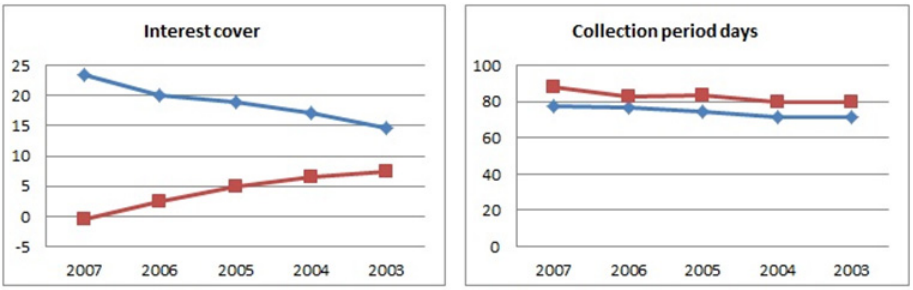


Fig. 1. An example of graphical comparison (blue = active, red = bankrupt): good resolution vs.bad resolution

with a good resolution for companies' separation from bankruptcy point of view. For this purpose a graphical comparison of their average values in time series was applied on the sample with medium size companies due to the highest number of both target cases (bankrupt or active), see Fig.1.

The new set of attributes included eight of the remaining attributesthat were further evaluated bymeans of correlation table. Only attributes with correlation lower than 0.5 have been selected for the modeling phase. As a result, five final financial indicators have been selected as input attributes for mining algorithms: net assets turnover (further denoted as A), interest cover (B), liquidity ratio (C), solvency ratio(D) and return on stakeholder funds (E).The final dataset was characterized by relatively balanced distribution for target attribute "company status in 2007" (55% active companies and 45% in bankruptcy).

2.3 Data Mining and Evaluation

Data mining (modeling phase) includes an application of suitable algorithms on preprocessed data to identify hidden knowledge, relations or dependencies. The whole modeling phase represents an iterative process with various modifications of input parameters based on obtained accuracy of the actual model and the goal is to find a model with the best prediction accuracy. We have used several machine learning algorithms represented further on by 3 types of models: decision trees, neural networks, and logistic regression.

The first experiment used a basic datasets for each company's category to identify potential for prediction models generation. For this purpose, one record in dataset was constructed as follows: ID, 2003 (A1, B1, C1, D1, E1), 2004 (A2, B2, C2, D2, E2), 2005 (A3, B3, C3, D3, E3), 2006 (A4, B4, C4, D4, E4), 2007 (Target). The aim of this experiment was to predict a future financial status of the companies based on their previous 5 years financial situation. Data partition was 70% for training and 30% for testing.

Results presented in the Table 2 show relative high accuracy for the third category of companies, but relevant confusion matrixes identified a high number of wrong classified records for target class "bankruptcy". This situation was caused by unbalanced distribution of target attribute (app. 20% records for bankruptcy and 80%

Table 2. Results of the 1st experiment

	Decision trees	Neural networks	Logistic regression (forwards)
Small	68.34%	63.31%	58.91%
Medium	72.78%	78.57%	54.23%
Large	76.98%	79.63%	78.84%

records for active large companies). The most plausible result was provided by neural networks and decision trees algorithms on medium companies' category and this fact motivated our decision to continue only with records from this category.

The second experiment used an extended dataset: the training set included values of five final indicators for years 2004, 2005, 2006 and 2007 with target attribute (company status) in 2007; the testing set included the same indicators, but for years 2005, 2006, 2007 and 2008. The objective was the same as in the previous experiment and achieved accuracy was very similar.

The last experiment was motivated by our effort to improve the accuracy whereas we wanted to preserve good resolving ability of the prediction models. At first we tried to extend a set of input attributes with additional financial indicators representing "Cashflow/operating revenue" and "Stock turnover". The new models based on decision trees and neural network algorithms resulted into accuracy less than 70 % with a worse separation in confusion matrix.

As the manual selection process didn't get a better accuracy, we tried an automatic selection method implemented in SPSS Clementine called feature selection to identify the most important attributes for target attribute "Company status". This method identified a list of twelve indicators(which included also the ones we used in previous experiments). The obtained results were similar as in the second experiment.Finally, a small improvement was achieved by controlled selection of records in the training sets to get a more balanced distribution of the target attribute. This operation improved the ability of algorithms (around 80%) to better predict a minor class (bankruptcy), see Table 3.

Table 3. Confusion matrix for combination of neural networks algorithm and medium size companies

		Actual class	
		Bankrupt (B)	Active (A)
Predicted class	B	1 116	110
	A	361	611

3 Business Intelligence

The second task deals with effective exploitation of selected open source BI solution to analyze Slovak and Czech companies from bankruptcy point of view. For this purpose, we designed and created a data warehouse containing extracted data from above mentioned databases within Vanilla BI[3] solution. The financial health status of investigated companies was evaluated by index IN_{99} designed by two well-known Czech economists Inka and IvanNeumaier [15] to analyze financial situation of enterprises in Czech Republic. Our motivation was to test its adaptability and suitability for very similar conditions in Slovak republic. The aim of this model is to identify if enterprise creates some Economic Value Added (EVA) or not, i.e. if performance of own capital is more than alternative costs for capital. The authors declare more than 98.9% probability of bankruptcy occurrence for cases with value losses and more than 84.6% for companies that create value.

The index gives a numerical value, which ranks the companies into one of the five categories:

1. $IN_{99} > 2{,}070$ company creates a value
2. $1{,}420 < IN_{99} < 2{,}070$ company mainly creates a value
3. $1{,}089 < IN_{99} < 1{,}420$ grey zone, it is not possible to determine if the company creates value or not.
4. $0{,}684 < IN_{99} < 1{,}089$ company mainly doesnot create a value
5. $IN_{99} < 0{,}684$ company doesnot create a value

The index calculation is as follows:

$$IN99=-0{,}17*X1+4{,}573*X2+0{,}481*X3+0{,}015*X4$$

X1 = Assets / Foreign sources
X2 = Profit before interest and taxes / Assets
X3 = Total returns / Assets
X4 = Current assets / Current liabilities, bank loans and current financial assistance

3.1 Data Description

Data for the data warehouse were extracted from two commercial databases, Albertina and Creditinfo, collecting information about registered companies and non-profit organizations in the Slovakia and Czech Republic for years 1992 till 2006. Albertina contains basic identification and contact information about companies and their owners, shareholders, partners and managers; plus information about a number of employees, type of sector, annual turnover, etc. Creditinfo Companies Monitor offers financial information about companies obtained from their accounting statements. We have used data only from 2001 to 2005 for our analysis, since this period contained the most numerous accounting statements in sufficient quality.

[3] https://launchpad.net/vanilla

3.2 Data Preparation

The initial aggregated dataset contained information about more than 600 000 companies and organizations registered in Slovak Republic, but the accounting statements were available only for 10 313 records. Also this dataset contained a lot of missing values, so we selected only 3 337 subjects with 12 613 accounting statements for data warehouse creation. This process started with basic ETL method including load of all records and attributes to Vanilla BI Gateway repository, see Figure 2. The imported data was processed again in order to identify crucial attributes and their quality, to eliminate redundancies and missing values caused by ambiguous duty for Slovak companies to publish their accounting statements.

The next step described in the following section covers a transformation of the cleaned data to the tables of dimensions and facts.

3.3 Data Warehouse: Design and Implementation

The used data warehouse was created through the basic star architecture (scheme) including one table of fact and eleven dimension tables connected by primary keys. The methodology for related attributes definition and their transformation into tables was carried out according to the mentioned scheme, see Fig.2.

The table of facts is denoted as "fct_fy" and consists of the following attributes:

- ICO – unique identification number of the company,
- AKTIVA_CELKOM - total assets of the company,
- HMOTNY_INV_MAJETOK –tangible fixed assets,
- OBEZNE_AKTIVA - current assets,
- FIN_MAJETOK - financial assets (short-term and long-term),
- PASIVA_CELKOM – liabilities,
- VL_IMANIE – equity,
- ZAKL_IMANIE –capital,
- CUDZIE_ZDROJE –foreign sources including short-term a long-term reserves, bank loans, etc.

The set of dimension tables contained eleven independent tables connected with the facts table using foreign keys:

- Dimension dim_pravna_forma – legal form of the company.
- Dimension dim_vlastnictvo – ownership of the company, e.g. cooperative; international in private sector; international in public sector; private domestic; state; in ownership of the state self-government, political parties or churches; and foreign ownership.
- Dimension dim_velkost – size of the company in terms of the employee number, e.g. micro-companies (0-9 employees), small companies (10-49 employees), middle-sized companies (50-249 employees) and large companies (more than 250 employees).
- Dimension dim_typ – type of the company, i.e. mainly trade oriented, mainly manufacturing or mainly oriented on providing services.

- Dimension dim_rok_vzniku – two levels: first level defines an exact date of the company origin; the higher level divides a time period before the year 1989 and after.
- Dimension dim_rok_uzavierky – year of financial statement publishing.
- Dimension dim_sidlo – information about the company's location.
- Dimension dim_okec – company's specialization based on economic activities categorization.
- Dimension dim_hosp_vysledok – profit or lost.
- Dimension dim_index$_{99}$ – two levels: classification into one of the 5 categoriesdescribed above and concrete calculated value of the index.
- Dimension dim_quick_test – two levels: different company's evaluation using 5 categories: very good, good, medium good, bad and at risk with concrete value on the second level.

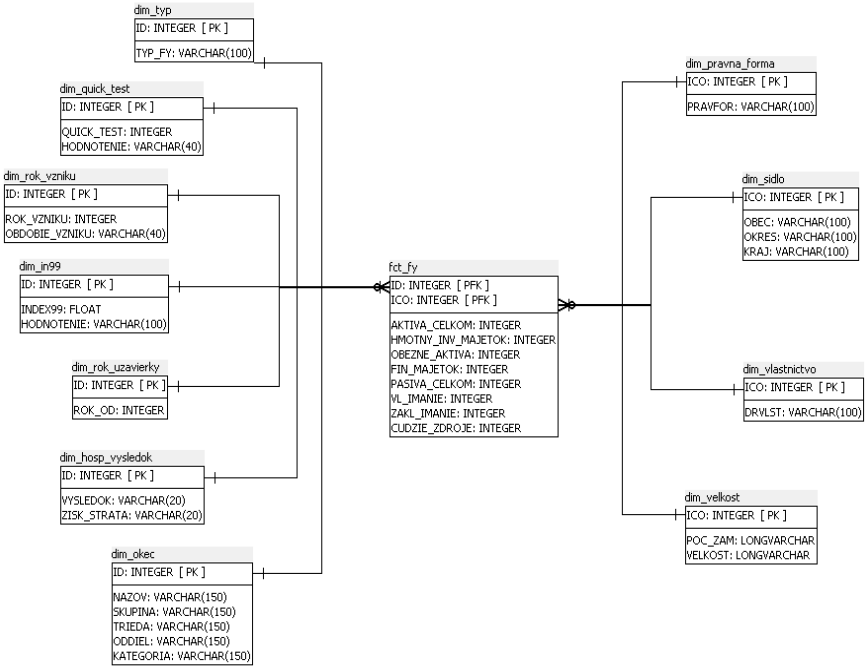


Fig. 2. Schema of created data warehouse

3.4 Analyses and Reports

BI solutions offer various analytical methods realized through OLAP operations over underlying data warehouse. In our case, a bankruptcy analysis was created within Free Analysis Schema Designer tool as integrated part of Vanilla BI solution. The calculated values of IN$_{99}$with final companies division into five categories is displayed on Figure 3.

The more detailed results can be obtained within a drill-down OLAP operation (dim_sidlo dimension) in order to identify a region in Slovakia with the highest number of companies in bankruptcy = Bratislava, the capital city of Slovakia. The next step to specify current result can be performed by adding dimensions dim_vlastnictvo and dim_type. In the final report each company is defined by its ICO and relevant combination of the values of selected attributes.

The implemented data warehouse and deployed BI solution offers a wide range of opportunities on how to analyze collected financial data. It is only necessary to specify the right OLAP operations and a way of visualizing the results.

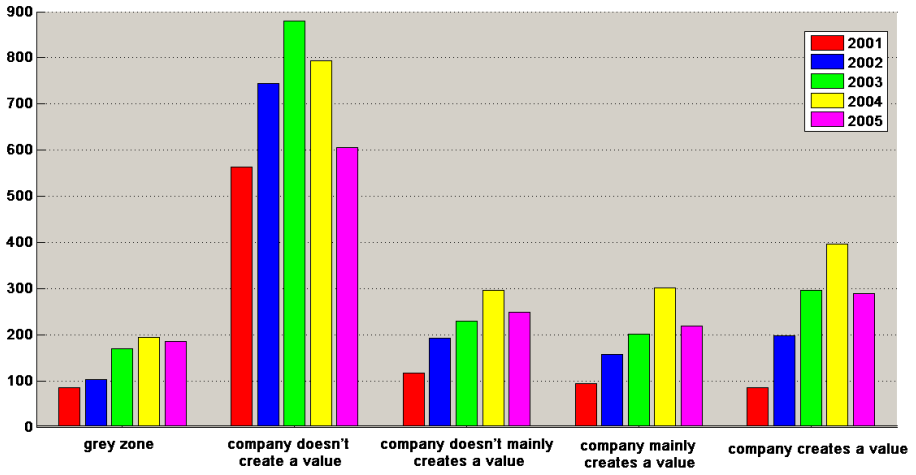


Fig. 3. Evaluation of Slovak companies from the viewpoint of bankruptcy

4 Conclusion

This paper presents two experimental studies how to analyze financial data within suitable methods of knowledge discovery and business intelligence for bankruptcy prediction. In the first case, we used extracted sample from financial database Amadeus to generate and evaluate several models for bankruptcy prediction. This approach included an interesting combination of selection process for crucial financial indicators (inputs for models) based on graphical comparison of their average values with selected mining algorithms for prediction models generation: decision trees, neural networks and logistic regression. The partial results inspired a discussion about what will be a worse scenario: to predict a bankruptcy and company will be still active or on the other hand to predict an active status of the company, when it goes bankrupt. From our point of view the consequences of the second scenario are much more negative than in the first one.

The second approach included the whole process of data warehouse creation with its further exploitation for analytical purposes. This analysis resulted in basic understandable and visualized results based on specified OLAP operations over

implemented data warehouse through selected open source solution. This approach confirms a high potential of open source BI applications to realize various OLAP operations in on-line form within user-friendly web environment. Obtained information can be used to support the decision and management processes on each company's level.

Presented results are plausible, but affected by several factors such as relatively high amount of missing values for financial indicators or unbalanced distribution of the target attribute. Impact of these factors motivates our future work with the following activities: aggregation of exported data from Amadeus database with additional data sources, an improvement of the indicators selection process with domain expert knowledge or other existing methods (financial, statistical, etc.); use different companies segmentation e.g. the one based on industry sectors, which has potential to get better results in line with relevant sector's characteristics; comparison of companies financial status before and after the global economic crisis.

Acknowledgments. The work presented in this paper was partially supported by the Slovak Grant Agency of Ministry of Education, Science, Research and Sport and Academy of Science of the Slovak Republic under grant No. 1/1147/12 (70%) and by the Slovak Cultural and Educational Grant Agency of the Ministry of Education, Science, Research and Sport of the Slovak Republic under grant No. 065TUKE-4/2011 (30%).

References

1. Altman, E.I.: Corporate Distress Prediction Models in A Turbulent Economic and Basel II Environment. Social Science Research Network, NYU Working Paper No. FIN-02-052, pp. 10–16 (2002)
2. Back, B., Laitinen, T., Sere, K.: Neural networks and bankruptcy prediction: Funds flows, accrual ratios, and accounting data. Advances in Accounting 14, 23–37 (1996)
3. Bertsekas, D.P., Tsitsiklis, J.N.: Neuro-dynamic programming. Athena Scientific (1996)
4. Bryant, S.M.: A case-based reasoning approach to bankruptcy prediction modeling. Intelligent Systems in Accounting, Finance and Management 6, 195–214 (1997)
5. Butka, P., Pócsová, J., Pócs, J.: Design and implementation of incremental algorithm for creation of generalized one-sided concept lattices. In: CINTI 2011: 12th IEEE International Symposium on Computational Intelligence and Informatics, Budapest, Hungary, pp. 373–378. IEEE (2011)
6. Butka, P., Pócs, J., Pócsová, J.: Use of Concept Lattices for Data Tables with Different Types of Attributes. Journal of Information and Organizational Sciences 36(1), 1–12 (2012)
7. Butka, P., Pócs, J., Pócsová, J., Sarnovský, M.: Multiple data tables processing via one-sided concept lattices. In: Zgrzywa, A., Choroś, K., Siemiński, A. (eds.) Multimedia and Internet Systems: Theory and Practice. AISC, vol. 183, pp. 89–98. Springer, Heidelberg (2013)
8. Hauser, R., Booth, D.: Predicting Bankruptcy with Robust Logistic Regression. Journal of Data Science (9), 565–584 (2011)

9. Kuznetsov, S.O.: Machine Learning and Formal Concept Analysis. In: Eklund, P. (ed.) ICFCA 2004. LNCS (LNAI), vol. 2961, pp. 287–312. Springer, Heidelberg (2004)
10. Madalla, G.S.: Introduction to Econometrics. Wiley, New York (2001)
11. Mahesh, A.R., Sivanandam, S.N.: Business Intelligence: Identify Valued Customer from the Data Warehouse in Financial Institutions. In: IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), India, pp. 1–5 (2010)
12. Mark, J., Goldberg, M.A.: Multiple Regression Analysis and Mass Assessment: A Review of the Issues. The Appraisal Journal, 89–109 (January 2001)
13. Martin, A., Manjula, M., Venkatesan, P.: A Business Intelligence Model to Predict Bankruptcy using Financial Domain Ontology with Association Rule Mining Algorithm. IJCSI International Journal of Computer Science Issues 8(3(2)), 211–218 (2011)
14. Martin, D.: Early warning of bank failure: A logit regression approach. Journal of Banking and Finance 1, 249–276 (1977)
15. Neumaier, I.: Index IN: Rychlý test kondice podniku. Ekonom Journal (13), 61–63 (2000)
16. Ohlson, J.A.: Financial rations and the probabilistic prediction of bankruptcy. Journal of Accounting Research 18, 109–131 (1980)
17. Pócs, J.: Note on generating fuzzy concept lattices via Galois connections. Information Sciences 185(1), 128–136 (2012)
18. Siramuthu, S., Ragavan, H., Shaw, M.J.: Using feature construction to improve the performance of the neural networks. Management Science 44(3) (1998)
19. Tam, K.Y.: Neural network models and the prediction of bank bankruptcy. Omega 19(5), 429–445 (1991)
20. Wilson, R.L., Sharda, R.: Bankruptcy prediction using neural networks. Decision Support Systems 11, 545–557 (1994)

Knowledge Compilation for Core Competence Extraction in Organizations

Simona Colucci[1], Eufemia Tinelli[2], Silvia Giannini[2], Eugenio Di Sciascio[2], and Francesco M. Donini[1]

[1] DISUCOM, Università della Tuscia, Viterbo, Italy
[2] SisInfLab & DEI, Politecnico di Bari, Bari, Italy

Abstract. Knowledge Management (KM) asks for information-intensive services over large amount of data, modeling the intellectual capital of an organization. To combine the expressiveness of logic-based languages with efficient information processing, we adopt a Knowledge Compilation approach to the extraction of "Core Competence" of a given company, a typical KM problem. In particular, we translate into a relational database schema the full logical description formalized in a Knowledge Base (KB), modeling organizational intellectual capital according to the formalism of Description Logics (DLs). Core Competence extraction is consequently performed through standard-SQL queries, while retaining the expressiveness of the logical representation. The service has been embedded in a system for Human Resource Management, `I.M.P.A.K.T.`[1], to show how Core Competence extraction performance significantly improves w.r.t. implementations exploiting DL reasoning engines.

Keywords: Description Logics, Core Competence Extraction, RDBMS, SQL.

1 Introduction

Knowledge management in an organizational environment is a complex, heterogeneous and information-intensive task which deserves the design of specific and sophisticated services in order to be performed. KM-related processes may in fact range in a wide set of tasks, from strategic choices decision support to business activities allocation, just to name a few. The automation of such processes is usually achieved by integrated enterprise suites relying on database technologies[2], which are able to cope with scalability issues, but – if traditionally employed – do not allow for a machine-understandable representation of the full informative content to be managed. On the contrary, the peculiarity of knowledge as organizational asset asks for technologies facilitating both representation and processing of information, like semantic-based ones [1]. In this paper we show how to combine the advantages of both semantic-based and database technologies in a Knowledge Compilation [2] approach, devoted to the identification of Core Competence in organizational support contexts.

Hamel and Prahlad [3] define Core Competence as a sort of organizational capability providing customer benefits, hard to be imitated from competitors and possessing

[1] Information Management and Processing with the Aid of Knowledge-based Technologies.
[2] `http://www.monster.com/`, `http://www.careerbuilder.com`

W. Abramowicz (Ed.): BIS 2013, LNBIP 157, pp. 163–174, 2013.

leverage potential. Among available approaches to strategic management (see [4] for a classification), our proposal takes the *competence-based perspective* ([5], [6]) and interprets company strategic competence as a collective asset, resulting from the synergy of human resources. We model the company intellectual capital in a Knowledge Base, which is described according to the formalism of DLs [7].

Coherently with the adopted Knowledge Compilation schema, our contribution makes computationally efficient Core Competence extraction over the information contained in the KB, by splitting the reasoning process in two phases: (i) the KB is pre-processed, thus parsing it in a specifically designed relational database schema (*off-line reasoning*); (ii) the querying process for Core Competence extraction is performed by exploiting the structure coming from the first phase (*on-line reasoning*). The approach has been implemented in `I.M.P.A.K.T.`, a system that manages skill matching [8] and team composition [9] tasks together with Core Competence extraction. `I.M.P.A.K.T.` implements via SQL queries standard plus non–monotonic and non–standard DLs inference services.

Other systems exploiting DBMS techniques to deal with reasoning tasks have been proposed in the literature (see *KAON2*[3], [10], [11], *QuOnto*[4] and *PelletDB*[5], among others). Even when their languages are more expressive than the one we use in our system, they are mostly able to return either exact matches (*i.e.*, instance retrieval) or general query answering. Instead, we use an enriched relational schema to deal with non-standard inferences to provide effective value-added services, including an approximate matching specifically fitting human resources management.

The paper is organized as follows: in the next Section we briefly recall both the DLs formalism and the adopted reasoning services, to make the paper self-contained. Section 3 details the Knowledge Compilation schema at the basis of the Core Competence extraction process, outlined in Section 4. The achieved performance improvements are presented in Section 5, through a comprehensive experimental evaluation. Finally, conclusions close the paper.

2 Background

Description Logics [7, Ch.2] are a family of formalisms and reasoning services for knowledge representation, whose alphabet is made up by unary and binary predicates, known as **Concept Names** and **Role Names**. Complex **Concept Descriptions** are built from concept and role names composed by *constructors*. Each choice of constructors defines a different DL, and characterizes it both in terms of expressiveness and computational complexity [7, Ch.3]. The expressiveness of a DL may be also enriched by the introduction of *concrete features*, which are binary predicates whose second argument belongs to a *concrete domain* D (*e.g.*, integers, reals, strings, dates). Given a DL $\mathcal{L}$, its enrichment with concrete features is denoted by $\mathcal{L}(\mathrm{D})$. Statements about classes in the domain of interest are divided into: **Concept Definitions** (denoted by $A \equiv C$) stating —in the form of a complex concept C—the necessary and sufficient conditions

[3] `http://kaon2.semanticweb.org/`
[4] `http://www.dis.uniroma1.it/~quonto/`
[5] `http://clarkparsia.com/pelletdb/`

for an individual to belong to the concept A; **Concept Inclusions** (denoted by $A \sqsubseteq C$) stating in C only the necessary conditions for membership in A. The set of inclusions and definitions formalize the intensional knowledge of the domain of interest, known as **TBox** in DL systems. A DL system usually allows one to make statements about named individuals, which make up the part of a DL-knowledge base known as **ABox**. **ABox** may include either **Concept assertions** ($C(a)$ states that an individual a belongs to the concept C) or **Role assertions** ($r(a, b)$ states that individual a relates to the individual b through role r).

In order to fully represent the features of Human Resources management, real-life examples suggest that at least the following constructors are needed: conjunction, universal and existential quantification, and concrete features (which define $\mathcal{ALE}(\mathrm{D})$). However, the interplay of existential and universal quantification leads to reasoning problems that are not computable in polynomial time [7, Ch.3], and such computational complexity hampers the translation into SQL of our problems.

Therefore, `I.M.P.A.K.T.` adopts a Curriculum Vitae (CV) representation (see Definition 2) allowing for reasoning only on $\mathcal{FL}_0(\mathrm{D})$ concepts. Such a DL drops existential quantification and thus gives up to the full $\mathcal{ALE}(\mathrm{D})$ expressiveness for reaching computability. The most important reasoning service in DL checks for specificity hierarchies, by determining whether a concept description is more specific than another one or, formally, if there is a *subsumption* relation between them. It is noteworthy that the framework underlying `I.M.P.A.K.T.` solves subsumption in $\mathcal{FL}_0(\mathrm{D})$ only via SQL queries, without reference to any exponential-time inference engine.

Although very useful in many knowledge management settings, subsumption does not allow to solve any emerging issue and non-standard reasoning services need to be specifically developed. In particular, the service category we here present aims at automatically extracting Core Competence, by identifying a common know-how in a significant portion of personnel, with such a portion cardinality to be set by the management. To this aim, our knowledge compilation approach framework follows the conceptual line in [12], based on Least Common Subsumer (LCS) computation.

By definition [13], for a collection of concept descriptions, their LCS represents the most specific concept description subsuming all of the elements of the collection. Nevertheless, in some applications, like Core Competence identification, such a sharing is not required to be full: the objective is finding a concept description subsuming *a portion of* the elements in a collection, rather than the collection as a whole. Such a concept description has been defined k-*Common Subsumer* (k-CS) [12]:

Definition 1. *Let $C_1, \ldots, C_n$ be n concepts in a DL $\mathcal{L}$, and let be $k < n$. A* ***k-Common Subsumer*** *(k-CS) of $C_1, \ldots, C_n$ is a concept D such that D is an LCS of k concepts among $C_1, \ldots, C_n$.*

Some k-Common Subsumers, defined *Informative k-Common Subsumers* ($IkCS$ [12]), are strictly subsumed by the collection LCS and then add informative content to it .

Among possible $IkCSs$, some subsume the biggest number of concepts in the collection and have therefore been defined as *Best Informative Common Subsumers* ($BICS$ [12]).

If a collection admits a meaningful LCS (*i.e.*, not equivalent to the universal concept $\top$), such LCS is the best common subsumer it may have. Else, for collections whose LCS is not meaningful, the *Best Common Subsumer* (BCS) has been defined [12].

3 Knowledge Compilation

I.M.P.A.K.T. takes all the information needed to model and manage the domain of human resources from a specifically developed modular ontology $\mathcal{T}$. The ontology currently includes nearly 5000 concepts modeling both the technical and the complementary competences an individual may hold.

In the following, we denote by $\mathcal{T} = \{M_i | 0 \leq i \leq 6\}$ the whole skills ontology adopted by the current implementation of I.M.P.A.K.T.. The ontology submodules M_i, with $i > 0$, are modeled according to $\mathcal{FL}_0(\mathrm{D})$ and describe different CV sections: Industry, ComplementarySkill, Level, Knowledge, Language, JobTitle. The ontology modularity allows for extending it whenever a new category of work-related features is identified, as shown below by the translation into a relational schema. Our default implementation of Core Competence extraction considers only Knowledge submodule, which models the hierarchy of possible candidate competence and technical tools usage ability; moreover, the module provides a type property to specify, for each competence, the related experience role (*e.g.*, developer, administrator, and so on) and two predicates: year to specify the experience level (in years), and lastdate which represents the last temporal update of work experience. M_0 is the main ontology module: it directly imports all the previous modules and models all of the properties needed for describing the candidate profile through the above detailed classes. We define as *entry points* such properties. In particular, M_0 includes one entry point for each imported sub-module.

Thanks to the knowledge modeling outlined so far, it is possible to model *CV Profiles* in the ABox. The CV classification approach we propose is based on a role-free ABox, which then includes only concept assertions of the form $P(a)$, stating that candidate a (*i.e.*, her CV description) offers profile features P (see Definition 2).

Definition 2. *Given the skill ontology $\mathcal{T}$, a **profile** P is a $\mathcal{ALE}(\mathrm{D})$ concept defined as a conjunction of existential quantifications, $P = \sqcap(\exists R_j^0.C)$, with $1 \leq j \leq 6$, where R_j^0 is an **entry point** and C is a concept in $\mathcal{FL}_0(\mathrm{D})$ modeled in M_j.*

As hinted before, our knowledge compilation problem aims at translating the skill knowledge base into a relational model, without loss of information and expressiveness w.r.t. our previous and fully logic-based work ([14]), in order to reduce on-line reasoning time. So, relational schema modeling is the most crucial design issue and it is strongly dependent on both knowledge expressiveness to be stored and reasoning to be provided over it. In particular, information storage involves both TBox axioms – concepts definitions, subsumption relations, value restrictions and profile descriptions – and ABox assertions – namely, profiles data represented as profile description instances – along with extra-ontological personal information. Notice that all non-standard reasoning services performed by I.M.P.A.K.T. process the atomic information making up the knowledge descriptions rather then the concept as a whole. For this reason, the availability of a finite normal form for such descriptions turns out to be very useful and

effective. We recall that $\mathcal{FL}_0(\mathrm{D})$ concepts can be normalized according to the *Concept-Centered Normal Form* (CCNF), [7, Ch.2].

According to such an approach, the KB is mapped to the database by means of the following design rules: **1)** a table CONCEPT is created to store all the atomic information managed by the system; **2)** three tables mapping recursive relationships over the table CONCEPT– namely PARENT, ANCESTOR and DESCONCEPT– are created; **3)** a table PROFILE includes the profile identifier ($profileID$ attribute) and the so called *structured information*: extra-ontological content, such as personal data (*e.g.*, last and first name, birth date) and work-related information (*e.g.*, preferred working hours, car availability); **4)** a table $R_j(X)$ is created for each entry point R_j^0, where $X = \{profileID, groupID, conceptID, value, lastdate\}$.

Given that for each conjunct $\exists R_j^0.C$ in $P(a)$, `I.M.P.A.K.T.` adds one tuple for each atom of the $CCNF(C)$ to the table $R_j(X)$, the attribute `groupID` is needed to convey all the informative content (*i.e.*, atoms of $CCNF(C)$) referred to the same conjunct. Thus, all features modeled in profile descriptions (Definition 2) are stored in tables $R_j(X)$ related to the involved entry points.

The relational schema resulting from the rules detailed so far allows for flexible skill matching classes, automated team composition, logic-based ranking and explanation of results (see [8] and [9] for more details). Here, we present formally only one match class, namely *Strict Match*, because it is at the basis of the development of *Core Competence extraction* in `I.M.P.A.K.T.`.

Definition 3. *Given the ontology $\mathcal{T}$, a set $\mathcal{FS} = \{fs_1, \ldots, fs_s\}$ of required candidate features, with each fs_i of the form $\exists R_j^0.C_i$, and a set $\mathcal{P} = \{P(a_1), \ldots, P(a_n)\}$ of candidate profiles, modeled according to Definition 2 and stored in the DB according to the schema detailed so far, the* ***Strict Match*** *process returns all the candidate profiles in $\mathcal{P}$ providing all the features fs_i in $\mathcal{FS}$.*

We notice that, thanks to CCNF, *Strict Match* can retrieve candidate profiles $P(a)$ more specific than $\mathcal{FS}$. If we think of $P(a)$ and $\mathcal{FS}$ in terms of their corresponding $\mathcal{ALE}(\mathrm{D})$ description (according to Definition 2), we may assert that *Strict Match* retrieves all the provided candidate Profiles ($P(a)$) linked by a subsumption relation to a target competence ($\mathcal{FS}$). Once the knowledge base $\mathcal{K}$ has been pre-processed and stored into the DB according to our relational schema, `I.M.P.A.K.T.` is able to perform all the reasoning services –also the *Strict Match* – only through standard SQL queries (see [8] for more details). From database querying point of view, fs_i has to be translated in a set of syntactic elements to search for in the proper $R_j(X)$ table. Then, for each fs_i one query $Q_s(fs_i)$ is automatically built on-the-fly considering a number of conditions in `WHERE` clause defined according to atoms in C_i. The final result of *Strict Match* is the intersection of profiles returned by all performed $Q_s(fs_i)$ queries. In the following, it is shown an executable example for the request $fs_i = \exists hasknowledge.(Java \sqcap \forall skillType.Programming \sqcap \geq_3 years)$ (the query has three conditions in `WHERE` clause, as the reader may easily expect).

```
SELECT profileID FROM hasKnowledge as R
WHERE conceptID = (SELECT conceptID
                   FROM concept WHERE name='Java')
AND EXISTS (SELECT * FROM hasKnowledge
            WHERE profileid=R.profileid AND groupid=R.groupid
```

```
            AND conceptid = (SELECT conceptID
                             FROM concept
                             WHERE name='skillType.Programming'))
AND EXISTS (SELECT * FROM hasKnowledge
            WHERE conceptid=(SELECT conceptID
                             FROM concept WHERE name='years')
            AND value >= 3 AND profileid=R.profileid
            AND groupid=R.groupid)
```

4 Core Competence Extraction

As hinted before, our proposal to Core Competence Extraction takes the competence-based perspective, which interprets company strategic competence as a collective asset, resulting from the synergy of human resources.

We observe that, even though a fully (as concerns both representation and reasoning) logic-based Knowledge Management System solving Core Competence extraction have been presented in [14], the novelty of our proposal lies in adopting a knowledge compilation approach, making the computational cost of the whole process to be affordable also for large organizations, while retaining the full expressiveness of the logic-based approach. Therefore, coherently with the novel knowledge compilation approach introduced in this paper, we show in the following how to solve Core Competence extraction through the *Strict Match* execution over the database schema presented in Section 3. To this aim, `I.M.P.A.K.T.` services rely on the computation of partial Common Subsumers defined in Section 2.

In [12] the *Common Subsumer Enumeration* algorithm was proposed, determining the sets $\underline{BICS}$, $\underline{CS_k}$, $\underline{ICS_k}$, $\underline{BCS}$ of, respectively, BICS, k-CS, IkCS, BCS of a collection $\{C_1, \ldots, C_n\}$, given $k < n$. The algorithm extracts from the set of profiles at hand the knowledge components shared by a significant number of individuals in the set, with such a significance level to be set as a threshold value (k) by the people in charge for strategic analysis. The algorithm works by taking as input a concept collection in the form of a *Subsumers Matrix* and the threshold value k.

`I.M.P.A.K.T.` implements the above recalled algorithm. Nevertheless, in order to cope with the features of the concept collection at hand, we here need to redefine a *Profiles Subsumers Matrix* (see Definition 5), and its preliminary Definition 4. We notice that, according to Definition 2, a Profile embeds all of the qualitative information detailed in the company CVs. Instead, in the current implementation, the identification of company Core Competence is limited to the evaluation of technical knowledge. Such an assumption affects the following definition of Profile Concept Components, defined w.r.t. a set EP of entry points R_j^0 of interest.

Definition 4. *Let P be a profile according to Definition 2, $P = \sqcap(\exists R_j^0.C)$, with C in $\mathcal{FL}_0(\mathrm{D})$ written in a CCNF $C^1 \sqcap \ldots \sqcap C^m$ and let EP be a set $EP \subseteq \{1, ...6\}$. For $j \in EP$, the **Profile Concept Components** (PCC) of P w.r.t. EP are defined as follows:*

1. if C^k, with $1 \leq k \leq m$, is a concept name, then $\exists R_j^0.C^k$ is a PCC of P;
2. if C^k, with $1 \leq k \leq m$, is a concrete feature, then the concept $\exists R_j^0.(C^k \sqcap C^f)$ is a PCC of P, for each $f \in \{1, \ldots m\}$ such that $f \neq k$ and C^f is a concept name;

3. for each C^k, if $C^k = \forall R.E$, with $1 \leq k \leq m$, then, for each E^h PCC of E, the deriving PCC of P are: i) $\exists R^0_j.\forall R.E^h$; ii) $\exists R^0_j.(\forall R.E^h \sqcap C^f)$, for each $f \in \{1, \ldots m\}$ such that $f \neq k$ and C^f is a concept name.

The identification of profile concept components is at the basis of the *Profiles Subsumers Matrix* construction, defined as follows.

Definition 5. *Let $\mathcal{P} = \{P(a_1), \ldots, P(a_n)\}$ be the set of profiles modeled according to Definition 2. Let now EP be a set such that $EP \subseteq \{1, ...6\}$, and $D_k \in \{D_1, \ldots, D_m\}$ be the* PCC *w.r.t. EP deriving from the collection $\mathcal{P}$. Given Definition 3, we define the **Profiles Subsumers Matrix** (PSM) $S = (s_{ik})$, with $1 \leq i \leq n$ and $1 \leq k \leq m$, such that: (i) $s_{ik} = 1$ if $P(a_i)$ strictly matches the component D_k; (ii) $s_{ik} = 0$ if $P(a_i)$ does not strictly match the component D_k.*

The above introduced characterization of the set $\mathcal{P}$ of the available knowledge profiles allows *Common Subsumer Enumeration* algorithm to retrieve the set of common subsumers useful to determine company Core Competence. In order to improve readability of the rest of the paper, the following definitions are provided.

Definition 6. *Referring to the PSM of the set $\mathcal{P} = \{P(a_1), \ldots, P(a_n)\}$, we define:*
***Concept Component Signature** (sig_{D_k}) the set of indexes of profiles $\{P(a_1), \ldots, P(a_n)\}$ strictly matching D_k (note that $sig_{D_k} \subseteq \{1, \ldots, n\}$);*
***Concept Component Cardinality** (T_{D_k}) the cardinality of the set sig_{D_k}, that is, how many profiles in $\mathcal{P}$ strictly match D_k. Such a number is $\sum_{i=1}^{n} s_{ik}$.*

5 System Performances

Core Competence extraction has been implemented as an enterprise business service for the `I.M.P.A.K.T.` system. It is a client-server application developed in Java exploiting Jena API to access the ontology model and Pellet reasoner to classify the ontology in the off-line pre-processing phase. Current implementation uses the open source PostgreSQL 9.1 DBMS.

In order to prove the effectiveness and efficacy of `I.M.P.A.K.T.` services, we initially created a real data set by collecting about 180 CVs, from three different employment agencies of candidates specifically skilled in ICT domain, so to simulate the scenario of an actual company in the ICT industry. The same dataset has been exploited for an iterative refinement phase of both the Skill Ontology development and testing of results of `I.M.P.A.K.T.` services.

Here, we are interested in the evaluation of *data complexity* and *expressiveness complexity* of our knowledge compilation approach to Core Competence extraction. To this aim, we perform two different test campaigns: the first one compares the performance of our implementation w.r.t. a fully logic-based one ([14]); the second one is focused on showing scalability capabilities. In each test category we adopted a different pair of datasets (namely, DS_1 and DS_2 in the first and DS_3 and DS_4 in the second category). Items in each mentioned pair differ in the level of specificity of the included profiles: the first dataset (DS_1, or, respectively DS_3) is more generic than the second one (DS_2, or, respectively DS_4) and thus characterized – for the same number of profiles – by a smaller set of resulting profile concept components.

For all the performed tests, `I.M.P.A.K.T.` was executed on an Intel Dual Core server, equipped with a 2.26 GHz processor and 4 GB RAM. Moreover, only CV information related to technical knowledge have been considered for Core Competence extraction (i.e. concepts of the form $\exists R_j^0.C$, where the entry point $R_j^0 = hasKnowledge$). We evaluate the two main extraction steps: the *Profile Subsumers Matrix* computation and the *Common Subsumer Enumeration* (CSE) algorithm execution (we recall that such algorithm takes the PSM as input). In the following, t_{psm} represents the average PSM computation time, t_{cse} represents the CSE time and n the number of profiles.

Figure 1 and Figure 2 show the performance results referring, for DS_1 and DS_2, to subsets of 5, 10, 15 and 20 profiles (characterized by the same cardinality as those in [14]). Such a different sets cardinality is aimed at investigating on how the execution time increases with the number of analyzed profiles. We also notice that, thanks to the adoption of the two datasets DS_1 and DS_2, it is possible to investigate on the relation between execution time and the number of profile concept components resulting from the dataset, when the number of profiles is given. Such a test strategy allows for assessing the impact of profiles complexity on the execution time. In particular, in Figure 1, we show t_{psm} (Figure 1(a)) and t_{cse} (Figure 1(b)), both in milliseconds, vs. n. Figure 2 presents the same computation times vs. the profile concept components number. Moreover, both Figure 2(a) and Figure 2(b) refer to the profile concept components deriving from DS_1 and DS_2. Intuitively, for each n value in Figure 2(a) and Figure 2(b), the smaller computation time value refers to DS_1 and the bigger to DS_2. In both experiments, k is set to 3.

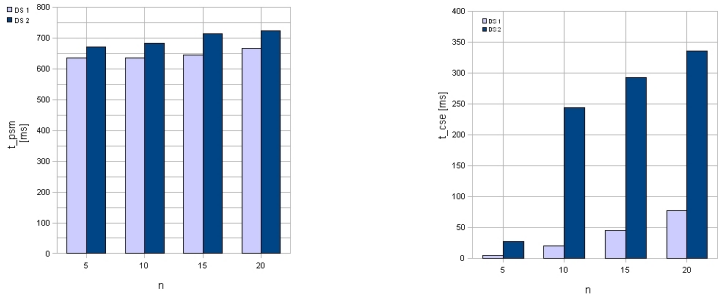
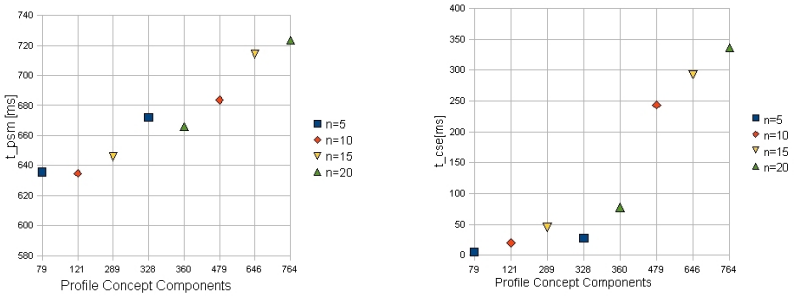


Fig. 1. PSM (a) and CSE (b) computation time vs. number of profiles

It can be noticed that the number of profiles is highly relevant in the common subsumer enumeration process, while the profile subsumers matrix computation time is less affected by such a number. As a general remark, t_{psm} is bigger than t_{cse}: the matrix creation is the most computationally expensive process. Nevertheless, by comparing presented matrix computation times with the ones shown in [14], the reader can notice how the adopted knowledge compilation approach dramatically improves process performance, by taking execution time from seconds to milliseconds (as an example, matrix computation time for 20 profiles has changed from 180 seconds to about 660 milliseconds), paving the way to the use of the approach in large real-world scenarios.



(a) (b)

Fig. 2. PSM (a) and CSE(b) computation time vs. profile concept components

For the second test campaign, aimed at evaluating the approach scalability, we carried out tests on pairs of synthetic datasets (subsets of the above mentioned DS_3 and DS_4), such that the number of features for each candidate is comparable to the average value of candidate profile features in the training real dataset. In particular, we implemented a KB instances generator, able to automatically create satisfiable profiles, according to Definition 2, also setting the features format (*i.e.*, number of features for each entry point, minimum number of specified technical skills, and so on). Thanks to the generator, we randomly created one dataset DS_3 of 500 profiles and extended it to 1000 and 2000 profiles. Then, we created a different dataset DS_4 of 500 profiles, by specifying a bigger average number of technical skills to be generated (30 instead of 3), so that the resulting profile concept components number increases (in the first case components arise up to 11643 for 2000 profiles, while in the second to 28177). Such a dataset has been extended to 1000 and 2000 profiles, too. In Table 1, the execution time for the two main steps of Core Competence extraction processes are shown w.r.t. DS_3 and DS_4 and $k = 0.3 \times n$.

Table 1. Core Competence extraction times (in seconds)

		Datasets Cardinality		
		500	1000	2000
t_{psm}	DS_3	0.87	1.1	1.74
	DS_4	3.91	9.24	27.29
t_{cse}	DS_3	0.21	0.46	1.4
	DS_4	66.06	235.81	912.57

It can be noticed that, if we consider DS_3, the most computationally expensive process is still the Profile Subsumers Matrix creation. On the contrary, in presence of significantly complex profiles – and consequently of a huge number of deriving concept components – the Core Competence Enumeration execution time dramatically raises (see values of t_{cse} referred to DS_4 and Figure 2(b)). As a consequence, we may hypothesize that, for an increasing number of concept components, there is a critical value after which the most time-consuming phase switches from the matrix computation to the common subsumers sets identification.

In order to provide a further rationale of the Common Subsumer Enumeration algorithm, we now show its results w.r.t. a subset of 8 CVs (out of the 180), modeled according to Definition 2 and stored in the DB according to the schema detailed in Section 3. The profiles example set is given hereafter, with reference only to technical knowledge (*i.e.*, $R_j^0 = hasKnowledge$):

Mario Rossi: Cplusplus (5 years), Java (5 years), Visual Basic(5 years)

Daniela Bianchi: Cplusplus (2 years), Java (6 years), Visual Basic (1 years)

Lucio Battista: DBMS (2 years)

Mariangela Porro: DBMS (2 years), Internet Technologies (2 years)

Nicola Marco: DBMS (5 years), Internet Technologies (5 years)

Carmelo Piccolo: VBScript, Process Performance Monitoring

Elena Pomarico: CplusPlus, Java, Visual Basic

Domenico De Palo: OOprogramming (6 years), Artificial intelligence (4 years), Internet technologies (4 years)

Ontology concept inclusions needed for understanding the proposed example are shown in the following:

$\texttt{ComputerScienceSkill} \sqsubseteq \texttt{EngineeringAndTechnologies} \sqsubseteq \texttt{Knowledge}$
$\texttt{ProgrammingLanguage} \sqsubseteq \texttt{SWDevelopment} \sqsubseteq \texttt{ComputerScienceSkill}$
$\texttt{OOP} \sqsubseteq \texttt{ProgrammingLanguage}$
$\texttt{VBScript} \sqsubseteq \texttt{ScriptLanguage} \sqsubseteq \texttt{ProgrammingLanguage}$
$\texttt{Java} \sqsubseteq \texttt{OOP}, \texttt{C++} \sqsubseteq \texttt{OOP}, \texttt{VisualBasic} \sqsubseteq \texttt{OOP}$
$\texttt{MySQL} \sqsubseteq \texttt{RDBMS} \sqsubseteq \texttt{OpenSourceDBMS} \sqsubseteq \texttt{DBMS} \sqsubseteq \texttt{InformationSystem} \sqsubseteq \texttt{ComputerScienceSkill}$
$\texttt{InternetTechnologies} \sqsubseteq \texttt{ComputerScienceSkill}$
$\texttt{ArtificialIntelligence} \sqsubseteq \texttt{ComputerScienceSkill}$
$\texttt{ProcessPerformanceMonitoring} \sqsubseteq \texttt{ManagerialSkill} \sqsubseteq \texttt{BusinessManagement} \sqsubseteq \texttt{Knowledge}$

In table 2, we sketch the structure of the profile subsumer matrix with reference to a subset of profile concept components (explained in Table 3), determined from the set $\mathcal{P}$ of 8 CVs according to Definition 4 (actual matrix dimension is 8×89).

Table 2. Portion of the example Profile Subsumers Matrix

	D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8	D_9	D_{10}	D_{11}	...
1	1	1	1	1	1	0	1	0	1	1	1	...
2	1	1	1	1	1	0	1	0	1	1	1	...
3	1	1	0	0	0	1	0	0	0	0	0	...
4	1	1	0	0	0	1	0	1	0	0	0	...
5	1	1	0	0	1	1	0	1	0	0	0	...
6	1	0	1	0	0	0	0	0	0	0	0	...
7	1	0	1	1	0	0	0	0	1	1	1	...
8	1	1	1	1	1	0	1	1	0	0	0	...

Table 3. Description of $D_1, \ldots, D_{11}$ reported in Table 2

D_1	$\exists$hasKnowledge.ComputerScienceSkill
D_2	$\exists$hasKnowledge.(ComputerScienceSkill$\sqcap$ $=_2$ years)
D_3	$\exists$hasKnowledge.ProgrammingLanguage
D_4	$\exists$hasKnowledge.OOP
D_5	$\exists$hasKnowledge.(ComputerScienceSkill$\sqcap$ $=_5$ years)
D_6	$\exists$hasKnowledge.(DBMS$\sqcap$ $=_2$ years)
D_7	$\exists$hasKnowledge.(OOP$\sqcap$ $=_5$ years)
D_8	$\exists$hasKnowledge.(InternetTechnologies$\sqcap$ $=_2$ years)
D_9	$\exists$hasKnowledge.C++
D_{10}	$\exists$hasKnowledge.VisualBasic
D_{11}	$\exists$hasKnowledge.Java
...	

According to the process introduced in Section 4, the system implements CSE algorithm and searches for profile concept components D_j whose cardinality T_{D_j} is greater or equal than a predefined threshold value k. In particular, k is set to the total number of profiles in the data set for the LCS computation. For the example set, the LCS computation returns: $\underline{LCS} = \{\exists\texttt{hasKnowledge.ComputerScienceSkill}\}$, which is quite a generic result for an ICT company.

The algorithm reveals also a slightly more significant information: the common know-how shared by the biggest portion of company personnel (but not by all of it) is: $\underline{BICS} = \{\exists \texttt{hasKnowledge}.\texttt{ComputerScienceSkill} \sqcap =_5 \texttt{years}\}$.

Finally, the process reveals that, for a degree of coverage set to 3, the set of company Core Competence includes all the following elements: $\underline{ICS}_3 = \{\exists \texttt{hasKnowledge}.(\texttt{DBMS} \sqcap =_2 \texttt{years}), \exists \texttt{hasKnowledge}.(\texttt{OOP} \sqcap =_5 \texttt{years}), \exists \texttt{hasKnowledge}.(\texttt{InternetTechnologies} \sqcap =_2 \texttt{years}), \exists \texttt{hasKnowledge}.(\texttt{C++} \sqcap \texttt{VisualBasic} \sqcap \texttt{Java})\}$.

In Figure 3, the I.M.P.A.K.T. Graphical User Interface (GUI) for the Core Competence extraction process is shown. Panel (a) provides the input user interface for choosing the degree of coverage k and the desired entry points to be considered in the extraction process. Panel (b) lists all possible pieces of company Core Competence, providing a user with the possibility to visualize (in panel (c)) the personnel holding such a strategic asset.

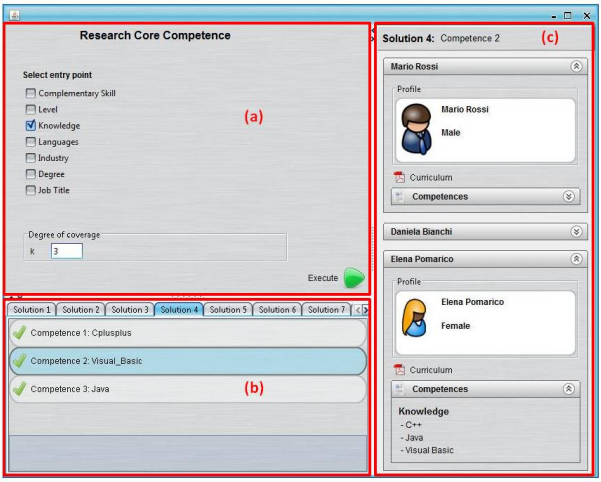


Fig. 3. I.M.P.A.K.T. GUI for Core Competence extraction

6 Conclusions

In the framework of I.M.P.A.K.T., an integrated system providing several knowledge management services, we showed how the process of company Core Competence extraction can be significantly improved in terms of performance by adopting a Knowledge Compilation approach. Thanks to such an approach, the full informative content modeled in a DL knowledge base has been translated in a relational database schema where advanced inference services can be executed exploiting standard-SQL queries. We showed, through experimental evaluation, the dramatic reduction obtained in terms of execution times with our novel approach. Implementation of database modeling denormalization and table partitioning are under way, and should further improve the performances.

Acknowledgements. We acknowledge support of projects UE ETCP "G.A.I.A." and Italian PON "VINCENTE - A Virtual collective INtelligenCe ENvironment to develop sustainable Technology Entrepreneurship ecosystems".

References

1. Draganidis, F., Mentzas, G.: Competency based management: A review of systems and approaches. Inform. Manag. and Computer Security 14(1), 51–64 (2006)
2. Cadoli, M., Donini, F.M.: A survey on knowledge compilation. AI Commun. 10(3-4), 137–150 (1997)
3. Hamel, G., Prahalad, C.K.: The core competence of the corporation. Harvard Business Review, 79–90 (May-June 1990)
4. Hafeez, K., Zhang, Y., Malak, N.: Core competence for sustainable competitive advantage: a structured methodology for identifying core competence. IEEE Transactions on Engineering Management 49(1), 28–35 (2002)
5. Tampoe, M.: Exploiting the core competences of your organization. Long Range Planning 27(4), 66–77 (1994)
6. Sanchez, R., Heene, A.: Reinventing strategic management: New theory and practice for competence-based competition. European Management Journal 15(3), 303–317 (1997)
7. Baader, F., Calvanese, D., Mc Guinness, D., Nardi, D., Patel-Schneider, P. (eds.): The Description Logic Handbook, 2nd edn. Cambridge University Press (2007)
8. Tinelli, E., Colucci, S., Giannini, S., Di Sciascio, E., Donini, F.M.: Large scale skill matching through knowledge compilation. In: Chen, L., Felfernig, A., Liu, J., Raś, Z.W. (eds.) ISMIS 2012. LNCS, vol. 7661, pp. 192–201. Springer, Heidelberg (2012)
9. Tinelli, E., Colucci, S., Di Sciascio, E., Donini, F.M.: Knowledge compilation for automated team composition exploiting standard SQL. In: Proc. of SAC 2012, pp. 1680–1685. ACM (2012)
10. Dolby, J., Fokoue, A., Kalyanpur, A., Schonberg, E., Srinivas, K.: Efficient Reasoning on Large SHIN Aboxes in Relational Databases. In: SSWS 2009, pp. 110–124. CEUR (2009)
11. Kiryakov, A., Ognyanov, D., Manov, D.: OWLIM – A pragmatic semantic repository for OWL. In: Dean, M., Guo, Y., Jun, W., Kaschek, R., Krishnaswamy, S., Pan, Z., Sheng, Q.Z. (eds.) WISE 2005 Workshops. LNCS, vol. 3807, pp. 182–192. Springer, Heidelberg (2005)
12. Colucci, S., Di Sciascio, E., Donini, F.M.: A knowledge based solution for core competence evaluation in human capital intensive companies. In: Proc. of I-KNOW 2008, pp. 259–266. Springer (2008)
13. Cohen, W.W., Borgida, A., Hirsh, H.: Computing least common subsumers in description logics. In: Proc. of AAAI 1992, pp. 754–760. AAAI Press (1992)
14. Colucci, S., Tinelli, E., Sciascio, E.D., Donini, F.M.: Automating competence management through non-standard reasoning. Engineering Applications of Artificial Intelligence 24(8), 1368–1384 (2011)

Stream-Based Recommendation for Enterprise Social Media Streams

Torsten Lunze[1], Philipp Katz[2], Dirk Röhrborn[1], and Alexander Schill[2]

[1] Communote GmbH, Dresden, Deutschland
torsten.lunze@communote.com
[2] Technische Universität Dresden, Fakultät Informatik, Deutschland
philipp.katz@tu-dresden.de

Abstract. Social media streams can be used for aggregating heterogeneous information sources into a single representation. In Enterprise Social Media Streams, employees interact with the stream and with other employees producing a constantly growing amount of new information. For avoiding an information overload, a recommendation engine must help the user to filter important information. This paper uses a *Stream Recommender System (SRS)* and presents an algorithm for an *SRS* to work within an enterprise context. The algorithms makes use of different social media specific features, including a feature that maintains a content-based user model. The algorithm has been evaluated against ratings, which have been collected within an existing productive Enterprise 2.0 system.

Keywords: Enterprise 2.0, Stream-based Recommender, Information Retrieval, Enterprise Social Media Streams.

1 Introduction

For public users as well as for knowledge workers within an enterprise, many different information sources exist. For helping a user to control the informations from various sources, systems exists that will aggregate these sources into one single stream. Such aggregation system are getting increasingly important for enterprise, especially if they follow the Enterprise 2.0 principle. Knowledge is shared and collaboration takes place within a system, such as a wiki, a microblogging tool, a forum, or various other system for storing and sharing messages and knowledge.

For building a single media stream, the following steps must be applied as described in our previous work [8] and in Figure 1: First, information must be obtained from the external systems using a set of specialized adapter components. Second, the acquired information must be transformed into a homogeneous format. Third, it must be annotated by using various information extraction methods. Fourth, a personalization algorithm must help to filter relevant information and fifth, the information must be presented to the user in an appealing and intuitive way.

W. Abramowicz (Ed.): BIS 2013, LNBIP 157, pp. 175–186, 2013.

Fig. 1. Processing Chain with focus of this paper marked red (modified version from [8])

Based on the previous work [8,9], this paper focuses on the personalization of messages as shown by the red rectangle in Figure 1. The goal of the personalization in an enterprise stream recommender is to reduce the number of information to consume by eliminating irrelevant information. Therefore it is is necessary to rank each message in nearly real-time, and not only to select the top n messages for recommendation. This is an important aspect which sets an enterprise recommender apart from well-known systems and algorithms employed for example in product recommendation engines. Also, such enterprise stream recommenders must be able to deal with permission and access levels. Typically, organizational structures like projects or departments can be defined within enterprises. This structure implies that individual employees have fine grained access privileges to specific resources. When sharing information between users or generated user models, it must be assured not to predict messages based on non accessible information.

This leads to the following definition: A **stream recommender** or **stream recommender system (SRS)** ranks items of a continuous stream at the moment as the item occurs in the stream using information that has been obtained during the past stream interaction. Therefore, an *SRS* must rank the items completely for each user who has access to the item.

The paper is organized as follows: In Section 2 related work is given. In Section 3, an algorithm for an *SRS* using social features is presented. Following this algorithm, an evaluation has been conducted in Section 4, and finally in Section 5, a conclusion is drawn.

2 Related Work

A recommender for social media streams is most similar to news recommenders, because they both mainly work on textual messages. In the most cases, news recommenders are using some form of content-based, collaborative and community-based technologies as in [5].

Comparing content-based with collaborative recommenders, [13] points out that content-based recommenders have their advantages in transparency, the handling of the so called “new item problem” and the user independence. Therefore, a plain collaborative recommender will not work since an *SRS* will have to

constantly recommend new items. Collaborative methods can only lead to acceptable results once enough relations between users and items can be derived.

Content-based news recommenders can be distinguished into term and concept weighting recommenders as in [12]. For an *SRS* mainly term weighting recommenders are relevant which applies common information filtering methods and applies them to learned user models.

[2,3] are using a stream recommender that works on the the twitter stream using content match and collaborative features. Also [1,15] describes how collaborative filtering can be used in a stream based scenario, mainly by adopting common algorithms to an iterative strategy.

The main are of news recommenders focus on public systems such as [4,11] and only a few target enterprise specific systems, such as [7,6,16,14]. However, non of those systems focus on ranking a whole stream to help the user to manage the information overload problem – only [14] showed a first approach, but lacks necessary performance and quality.

This paper closes the gap by introducing an algorithm for an *SRS* to rank new messages in near real-time and which is applicable within an enterprise environment.

3 Stream Recommendation Algorithm

As mentioned before, an *SRS* must be able to rank messages in real-time, honour the access levels on messages and help to hide irrelevant messages. To reach a near real-time ranking performance, the ranking process must be as efficient as possible and should mainly use precomputed data, such as a user model reflecting the individual users' interests.

The access constraint is solved by defining message groups: A message group can be seen as a project, and each message group has a set of user assigned who are able to read or write messages. The message group and access level structure is not defined within the recommender itself but obtained from the (social media) hosting system. During the processing, each incoming message can be easily identified as part of one or more message group, actually the message group is an attribute of the message. Once the recommender will use learned user models to make predictions for other similar users, the recommendation should not use terms of messages the user will not have access to. This should avoid predictions that may reveal sensible information.

3.1 Feature Definition

When computing a score for a new messages, the available data about social interactions with the message or related messages that can be exploited are used. This leads to the following feature definitions for a message m and a user u:

Root Feature. The current message m is the first message in a discussion.
Author Feature. The user u is the author of the message m.

Mention Feature. The user is mentioned in the message.
Discussion Participation Feature. The message is part of a discussion. The user is the author of another message within the discussion.
Discussion Notification Feature. The message is part of a discussion and the user has been mentioned in another message within the discussion.
Content Match Feature (CMF). The message is compared with a learned user model.

All of the above features except the *CMF* are boolean features that can be easily computed. They take values of either 0 or 1 for $false$ and $true$, respectively. The value of the *CMF* feature ranges from 0 to 1 denoting a similarity of the specific message to the user's model. The employed user model is fairly simple, it consists of two values for each term t:

- Sum of all rating values of user u for messages containing the term t.
- Number of ratings of user u that refers to a message that contains term t.

Based on a set of ratings, the user model can be learned incrementally by recording ratings from the user. The simplest way is to obtain explicit ratings given by the specific user. However, systems that only rely on explicit ratings are often not well accepted by the end user. An unintrusive way to obtain ratings and learn a user model is to use the features described above themselves. Preliminary evaluations showed, that the *Mention, Discussion Participation* or *Discussion Notification* Features for a user u and a message m correlate directly with the actual interest of the user in this message. Hence, when it is observed that one of the features applies to a message, the message is assumed as being relevant for the user and a positive rating is generated, triggering an update of the user model. The validity of this assumption will be confirmed in Section 4.

The ratio of both values is the score $um_u(t)$ for the term t in the user model u. With $rm(u)$ as the set of all messages for user u with positive ratings, and with $r(u, m)$ as the obtained rating for user u for message m the score can be expressed as follows:

$$um_u(t) = \frac{\Sigma_{m \in rm(u) \cap t \in m} r(u, m)}{\Sigma_{m \in rm(u) \cap t \in m}} \tag{1}$$

The final *CMF* feature score for a message m and a user u can then be easily computed using the average user model score as follows:

$$CMF(u, m) = \frac{\Sigma_{t \in m} um_u(t)}{\Sigma_{t \in m \cap t \in um_u}} \tag{2}$$

3.2 Ranking Process

The ranking process is visualized in Figure 2. The process is triggered when receiving a new message. In the first step, an information extraction takes place

that will extract terms from the messages as follows: First, the message is cleaned from HTML tags and other unparseable characters, second, the language of the message is detected and third, stemmed tokens are extracted based on the detected language. This is a fairly simple extraction but it can be easily extended in further versions.

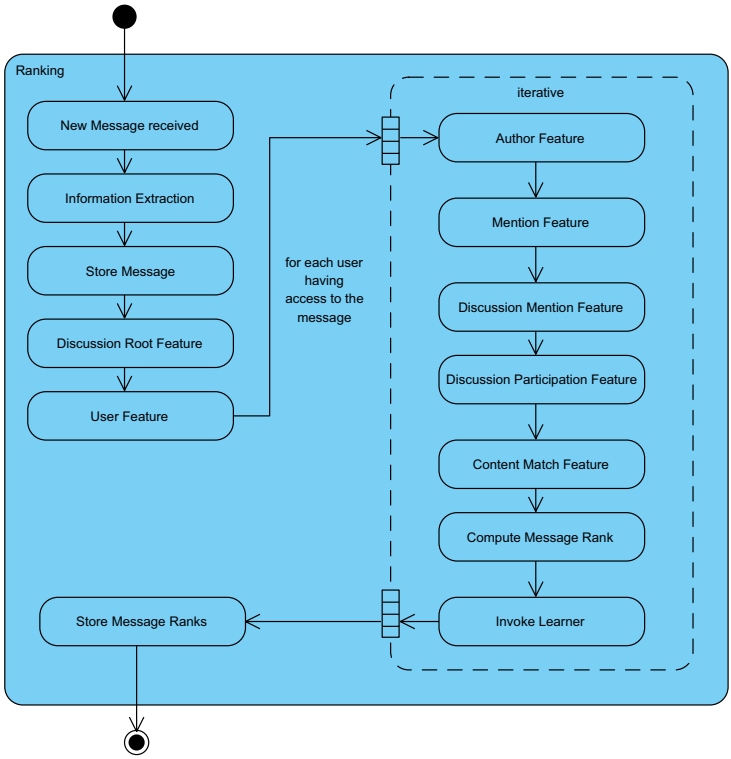


Fig. 2. This UML activity diagram shows the ranking process upon receiving a new message

The result of the information extraction is a set of terms that will be stored with the message. Then the features are computed per user and message. The *Discussion Root* Feature is user independent and can be computed before. The User Feature evaluates the access privileges for the message and then initiates the computation of the user dependent features for each user having access to the message.

The first user dependent feature is the *Author Feature* which determines, if the current user is the author of the message. The second feature is the *Mention Feature*, which determines if the user is mentioned within the message. Next, the *Discussion Participation Feature* checks, if the message is part of a discussion – or has related messages – and if the user is the author of one of the related message.

Similarly, the *Discussion Mention Feature* checks, if the author is mentioned within a message related to the current message.

The last feature to compute is the *Content Match Feature*, which matches the terms of the message against the user model of the user as defined in Equation 2.

After all features have been computed, the values of those features will combined into one score for the message. Currently, the following combination strategy is used:

1. If the *Author Feature* is *true*, the score is set to 1.
2. Else if the *Mention Feature* is *true*, the score is set to 0.95.
3. Else if the *Discussion Participation Feature* is *true*, the score is set to 0.9.
4. Else if the *Discussion Mention Feature* is *true* the score is set to 0.8.
5. Else the *Content Match Feature*'s value is used as a score.

The values have been based on a correlation analysis of the evaluation data set mentioned later. The details are skipped here due to space reasons.

Once the final message score is available, the learning process can be invoked directly as mentioned before. If either the *Author Feature*, *Mention Feature*, *Discussion Participation Feature* or *Discussion Mention Feature* is true, the learning process will be invoked by generating an observation with the computed score as interest value.

The ranking process ends by taking the computed scores and storing them for future usage – such as filtering or visualization.

3.3 Learning Process

In Figure 3, the learning process is visualized. The trigger for the process is a new observation: An observation is defined by a message, a user and a value representing the interest. The interest value represents the interest of the user in the message and a number in the interval $[0 \dots 1]$.

An observation can be made in the following cases:

1. The user rates a message explicitly.
2. A new message occurs, that has been identified as interesting for the user during the ranking process (see Section 3.2).
3. An interaction of the user with the system is identified (e. g. the user likes a message).

Once the learning process is initiated, the information extraction will run for the associated message if necessary. After the terms have been extracted, the user model entries for each term are updated as given in Equation 1.

For example, assume a user Lara has started a discussion and a new message m appears that belongs to that discussion. The *Discussion Participation Feature* is identified and will lead to a high score for Lara during the ranking process. Furthermore case 2 above applies and the terms of the message are used to update the user model of Lara.

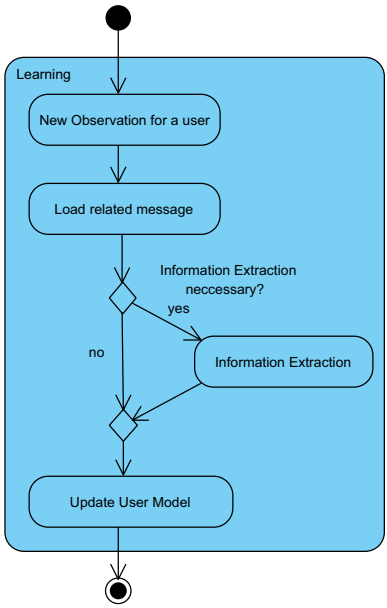


Fig. 3. This UML activity diagram shows the learning process

4 Evaluation

For the evaluation of the algorithm, Communote[1] has been used as Enterprise Social Media Stream Application. The core implementation of the *SRS* algorithm has been implemented in Java and is available as an Open Source project[2].

The productive installation of Communote within Communardo Software GmbH has been extended by the possibility to rank messages and filter for the computed score as shown in Figure 4. The user has the possibility to either mark a message as being relevant, not relevant or as undecided. In this evaluation run 9 users participated and they submitted a total of 11,160 ratings in January 2013 [3].

Prior to an evaluation run, the dataset has been split into a training set (70% of the ratings) and a test data set. The training set has been used to train the algorithm, that is to learn the user model of the *CMF*. Each message – even if no rating existed – was passed in temporal order to the ranking process, starting with the oldest message. If a rating in the trainings dataset was available for a message, it was passed to the learning process directly after the ranking process finished for the message. Also, if a rating for the message was available in the test dataset, it was evaluated directly after the learning processed finished.

[1] `http://www.communote.com`
[2] `http://www.spektrumprojekt.de`
[3] Unfortunately the dataset cannot be made public due to confidentiality reasons.

Fig. 4. Frontend for Ranking and Filtering Messages

This way, the order of the messages within the stream and the order of the ratings was taken care of to simulate the real world behaviour as well as possible.

Based on the obtained dataset, the following evaluation configurations have been conducted to analyse the behaviour and the quality of the algorithm:

All Features. All features (see Section 3.1) are used to predict a ranking value.
All Features. with Message-Group-specific CMF All features have been used to predict a ranking value. The *CMF* uses a separate user model per user per message group (project).
Only CMF. Only the *CMF* is used to predict the ranking value.
Only Message-Group-specific CMF. Only the *CMF* is used to predict the ranking value. The CMF uses a separate user model per user per message group (project).
No CMF. All features but the *CMF* are used to predict a ranking value.

For each configuration the precision, recall and the F_1- and F_2-scores have been evaluated. We decided to also use the F_2-score for evaluation since it sets more priority on the recall as on the precision. Hence, the score favours more to get all relevant information instead of missing a relevant information for higher precision.

The algorithm predicts a ranking value per message and user. We used different thresholds for deciding whether the prediction is *positive* (for being relevant to the user) or *negative* (for being irrelevant, respectively) to get a better understanding of the algorithm and the underlying dataset. The graphs for the different configurations are shown in Figure 5 to 9. The x-axis represents the threshold of predicting the algorithm output as *positive* and y-axis represents the value of the precision, recall, F_1- and F_2-score.

In Figure 5 and 6, all features have been used to compute predictions. In Figure 5 a global user model per user was used, in Figure 6 a specific user model per message group was used. As it can been seen, there is no significant difference regarding performance of both configurations.

When the *CMF* feature is not used, the recall drops significantly, however the other features prove to lead to a high precision as shown in Figure 7. The values are mostly constant in this case, since all of the used features are boolean and independent from the learned user model.

Finally, in Figure 8 and 9 only the *CMF* has been evaluated. It shows a similar behaviour as in using all features. Only at higher thresholds, the recall and hence the F-scores drop faster as in the case employing all features.

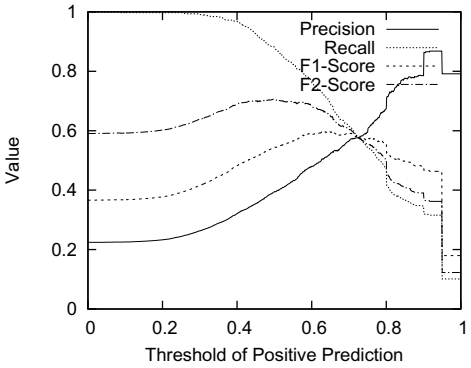


Fig. 5. Precision and recall curves for all features with a global user model per user

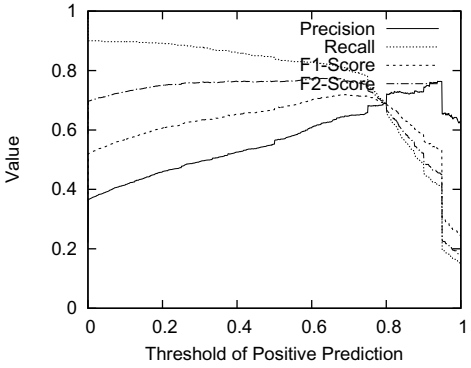


Fig. 6. Precision and recall curves for all features with Message-Group-specific user models

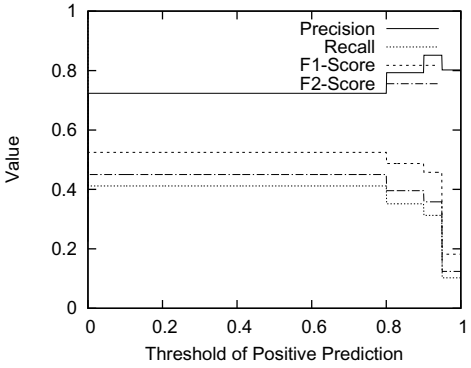


Fig. 7. Precision and recall curve for all features but the *CMF*

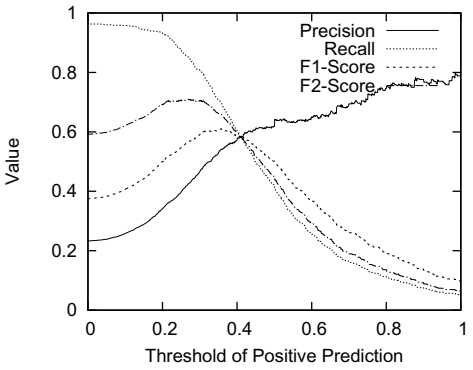


Fig. 8. Precision and recall curves for only the *CMF* feature with one global user model per user

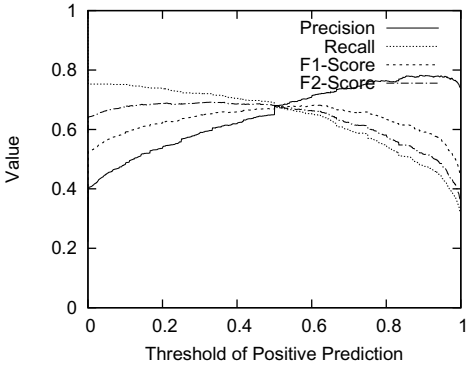


Fig. 9. Precision and recall curves for only the *CMF* with Message-Group-specific user models

Table 1. Maximum F1- and F2-scores for different configurations

Configuration	Max. F_1-score	Max. F_2-score
All Features	0.60	0.69
All Features with Message-Group-specific CMF	0.72	0.74
Only CMF	0.61	0.70
Only Message-Group-specific CMF	0.68	0.69
No CMF	0.52	0.59

The maximum achieved F-scores for each configuration are shown in Table 1. Based on those numbers, using Message-Group-specific user models lead to a significant better performance as a global user model. The reason is the organisation of the message groups as projects, and that similar content in different projects is of different interest for the user. Also the *CMF* itself leads to similar results in comparison to using all features, however the best value is reached by using all features.

5 Conclusion and Further Work

In this paper, an algorithm for an *SRS* has been introduced and evaluated on a real world dataset within an enterprise. The separation of the user model into message groups did not show a significant loss of quality. Therefore it is applicable to separate the user model for access level and recommendation purposes.

The evaluation showed that the features without the *CMF* lead to a high precision but they fail to identify important messages. Hence the combination of all the features is feasible, as the usage of the features for learning the user model.

The evaluation showed that the non content-based features can serve as a good indicator for relevant information, but they fail to predict all significant information. Further work will analyse the behaviour of the features as well as using other new features. For example, users can like messages, and once a user liked a message, a new observation can be made. Another part of future work is to analyse the *CMF* for improving the overall result, especially during a long term usage of the system. One idea is to use models of similar users to make a prediction in case the own user models fails to give a confident prediction.

Another extension is to identify project roles and use them as a feature. For example a project leader will have different interests within the same project as a supporting co-worker.

Acknowledgements. The results presented in this paper have been developed within the research project *SPEKTRUM*. This project is funded by the Free State of Saxony and the EU (European Regional Development Fund). We would like to thank all the users at the Communardo Software GmbH participating in creating the dataset for the evaluation.

References

1. Chandramouli, B., Levandoski, J.J., Eldawy, A., Mokbel, M.F.: StreamRec: a real-time recommender system. In: Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data, SIGMOD 2011 (2011)
2. Chen, J., Nairn, R., Nelson, L., Bernstein, L.E., Chi, E.: Short and tweet: experiments on recommending content from information streams. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI 2010, pp. 1185–1194 (2010)
3. Chen, J., Nairn, R., Chi, E.: Speak little and well: recommending conversations in online social streams. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI 2011, pp. 217–226 (2011)
4. Das, A., Datar, M., Garg, A., Rajaram, S.: Google News Personalization: Scalable Online Collaborative Filtering. In: Proceedings of the 16th International Conference on World Wide Web, p. 271 (2007)
5. Diaz-Aviles, E., Drumond, L., Schmidt-Thieme, L., Nejdl, W.: Real-Time Top-N Recommendation in Social Streams. In: Proceedings of the Sixth ACM Conference on Recommender Systems, p. 59 (2012)

6. Guy, I., Ronen, I., Raviv, A.: Personalized Activity Streams: Sifting Through the "River of News". In: Proceedings of the Fifth ACM Conference on Recommender Systems, p. 181 (2011)
7. Guy, I., Zwerdling, N., Ronen, I., Carmel, D., Uziel, E.: Social Media Recommendation based on People and Tags. In: Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, p. 194 (2010)
8. Katz, P., Lunze, T., Feldmann, M., Röhrborn, D., Schill, A.: System Architecture for handling the Information Overload in Enterprise Information Aggregation Systems. In: Abramowicz, W. (ed.) BIS 2011. LNBIP, vol. 87, pp. 148–159. Springer, Heidelberg (2011)
9. Katz, P., Feldmann, M., Lunze, T., Sprenger, S., Schill, A.: Authoring Processing Chains for Stream-based Internet Information Retrieval Systems. In: Abramowicz, W., Kriksciuniene, D., Sakalauskas, V. (eds.) BIS 2012. LNBIP, vol. 117, pp. 189–200. Springer, Heidelberg (2012)
10. Kim, Y.S., Yum, B.-J.: Recommender system based on click stream data using association rule mining. Expert Syst. Appl. 38(10) (September 2011)
11. Li, L., Wang, D., Li, T., Knox, D., Padmanabhan, B.: SCENE: a scalable two-stage personalized news recommendation system. In: Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, p. 125 (2011)
12. Li, L., Wang, D., Zhu, S., Li, T.: Personalized News Recommendation: A Review and an Experimental Investigation. Journal of Computer Science and Technology, 754 (2011)
13. Lops, P., Gemmis, M., Semeraro, G.: Content-based Recommender Systems: State of the Art and Trends. In: Recommender Systems Handbook, p. 73 (2011)
14. Lunze, T., Feldmann, M., Eixner, T., Canbolat, S., Schill, A.: Aggregation, Filterung und Visualisierung von Nachrichten aus heterogenen Quellen – Ein System für den unternehmensinternen Einsatz. In: Proceedings of the GeNeMe 2009 Workshop, Dresden (2009)
15. Nasraoui, O., Cerwinske, J., Rojas, C., Gonzalez, F.: Collaborative filtering in dynamic usage environments. In: Proceedings of the 15th ACM International Conference on Information and Knowledge Management, CIKM 2006, pp. 794–795 (2006)
16. Schirru, R., Baumann, S., Memmel, M., Dengel, A.: Topic-Based Recommendations for Enterprise 2.0 Resource Sharing Platforms. In: König, A., Dengel, A., Hinkelmann, K., Kise, K., Howlett, R.J., Jain, L.C. (eds.) KES 2011, Part I. LNCS, vol. 6881, pp. 495–504. Springer, Heidelberg (2011)
17. Wan, Y., Chen, C.: An Effective Cold Start Recommendation Method Using A Web of Trust. In: PACIS 2011 Proceedings (2011)

IT Audit Management Architecture and Process Model

Tiago Rosário, Rúben Pereira, and Miguel Mira da Silva

Department of Computer Science, Instituto Superior Técnico, Lisbon, Portugal
{tiago.rosario,rubenfspereira,mms}@ist.utl.pt

Abstract. Over the last few decades various regulations emerged and an auditor is the last line of defence against the risks of non compliance. Therefore, Information Technology (IT) Audit Management (AM) is a crucial process for most organizations. However, it is a complex process and current IT frameworks are not helping since they are seen as complex, overlapping each other, and hard to implement. The main contribution of this research is a formal and complete IT AM Process/architecture, useful and adaptable to all type of organizations, which is based on most relevant IT best practices frameworks, literature of the area and in practioners' viewpoint. The research methodology used was the Design Science Research (DSR). To evaluate our proposal we interviewed IT audit experts in order to add practitioners' perspective to it. We finish our research by providing the main contributions, limitations, and future work.

Keywords: IT Audit Process, Model, Architecture, BPMN, IT Frameworks.

1 Introduction

Over the last few decades, numerous organizations suffered financial losses, law suits, etc [1]. The occurrence of these scandals adversely impacted business and rudely awakened organizations to act [2]. Nowadays, with the financial crisis, the need for rigorous controls is rising [3]. For each new law or regulation, compliance departments need to design new internal policies and procedures to deal with the rule specifications [4]. However, there is no guarantee that all entities meet organizations requirements [5], and an auditor is the last line of defence in detecting problems [6].

Nevertheless, with the arrival of the information age, the impact of IT in organizations keeps growing [6] and IT began to be comprised in the organization's business core processes [7]. So, it is crucial to achieve a good alignment of IT with business needs [8], which increases the need for more requirements in this area [9].

Currently organizations are facing an increasing number of regulations (requirements) they need to be compliant with [1][5]. Plus, IT audit procedures have also become more complex [8]. So the way audits are performed is affected [2], IT auditors' effort is growing [13] and the degree of compliance achieved is decreasing [1].

Given the importance of IT audit, many IT frameworks have been developed to help organizations in IT AM process. However, IT frameworks are seen as complex [14], overlapping each other [15][16], hard to implement [17] and separately they do not propose a complete IT AM process [10]. As a result, organizations cannot implement a complete IT AM process [10]. Therefore, there is space for new and

W. Abramowicz (Ed.): BIS 2013, LNBIP 157, pp. 187–198, 2013.

innovative proposals regarding IT AM process/architecture topic. Therefore, the problem that this research intends to help solve is:

> *IT audit management is a crucial process for most organizations. However, it is a complex process and current IT frameworks are not helping since they are seen as complex, overlapping each other, and hard to implement.*

Since the definition of formal procedures to perform IT audits can bring benefits to organizations [1], and knowing that IT has become crucial to the support and growth of the business [11], in this research we propose to model the IT AM process/architecture taking into consideration the most relevant IT frameworks, literature of the area and practitioners' viewpoint. To model the IT AM process we used the Business Process Model Notation (BPMN), considered a de-facto standard for business process modelling [12].

2 Research Methodology

In contrast with behaviour research, design-oriented research builds a "to-be" conception and posteriorly seeks to build the system according to the defined model taking into account restrictions and limitations [18]. Design science addresses research through the building and evaluation of artefacts designed to meet identified business needs [19] instead of analysing existing Information Systems (IS) in order to identify causal relations [18].

As advised by March & Smith [20] the research methodology applied is divided according to the two processes of DSR in IS: build and evaluate. The build process is composed by two stages and the evaluation process is comprised by only one (Table 1). Based on the four design artefacts produced by DSR in IS (constructs, models, methods and instantiations) we will focus on constructs and models. This kind of research approach was already used in other researches [21].

In order to leverage the IT audit sub-phases, IT audit roles, IT audit activities, IT audit data and IT AM process information entities we will use extensive literature review. The approach used in this research follows the concept-centric methodology of IS literature reviews as outlined in [7].

Table 1. Research Methodology

Build				**Evaluate**
Constructs Definition	IT AM Process Construction	IT AM Information Architecture Construction	IT AM Information Systems Architecture Construction	Evaluation
IT Audit Sub-Phases1 IT Audit Roles IT Audit Activities IT Audit Data	Analyze the relationship between constructs and integrate constructs			Interviews Questionnaire
IT AM Process Information Entities		Analyze the relationship between constructs and integrate constructs		Interviews Questionnaire
IT AM Process IT AM Information Architecture			Analyze the relationship between constructs and integrate constructs	Interviews Questionnaire

Additionally, we followed the guidelines for DSR proposed by Hevner [19]. A design artefact is complete and effective when it satisfies the requirements and constraints of the problem that it was meant to solve. In this research we evaluated our artefacts through interviews and questionnaires. By submitting this research to respected international conferences, we also used the appraisal of the scientific community as evaluation criteria.

3 Related Work

Audits are conducted in diverse legal and cultural environments [1], within organizations that vary in purpose, size, complexity, and structure, and by people within or outside the organization [22]. Plus, audit is an independent and objective assurance activity [23] that employs systemized and standardized methods [1][24] to evaluate and improve the process of governance, risk management, control and treatment, so as to help the organization achieve its objectives. Notwithstanding, it is important to note that the role of audit management is not just to perform audits (Table 2).

Table 2. Audit aims

Audits provides:	References
Assurance	[1][2]
Assessment and Recommendations	[2][24]
Oversight	[24]
Advisry Services	[24]

Besides, to understand the complexity around Audit we present Rosário's conceptual map [10] that organizes and clarify all the issues about audit concept (Figure 1).

IT audit represents a procedure used to assess whether the IT acts in the function are successfully accomplishing the business objectives. IT audit also includes the use of IT to support audits [2] which allows for more efficient ways of analyzing the effectiveness of the implemented controls [1][25].

The IT audit definition is given by the Institute of Internal Auditors (IIA):"IT audit is the process of gathering and evaluating evidence based on which one can evaluate

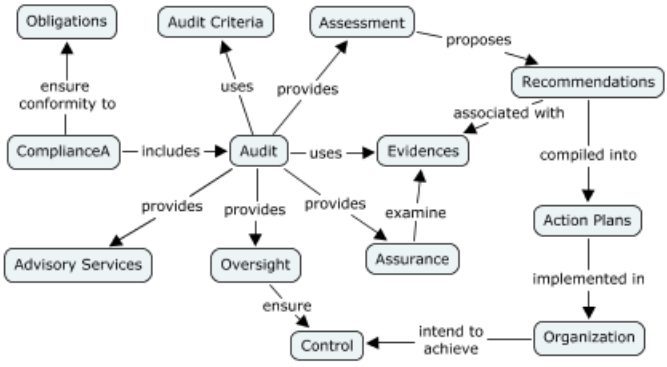


Fig. 1. Audit Conceptual Map. *Adapted from [10]*

the performance of IT systems, i.e. to determine whether the operation of IS in the function of preserving the property and maintain data integrity" [22].

The process can be described as a set of steps separated into phases, each one with a well-defined purpose to achieve audit goals [2]. There is a consensus about more generic audit phases [10] that constitute a one direction flux [8] (Figure 2). Due to space limitations we won't describe each phase but more information can be seen at [11][23][26].

Fig. 2. IT Audit Phases

4 Proposal

This research proposes the formalization of the IT AM process. In this section, we design and present the IT AM process and the IT AM architecture composed by the IT AM information architecture and IT AM information systems architecture by analyzing the most relevant IT frameworks and literature and eliciting information about how to perform audits. Using IT frameworks and literature's best practices recommendations we can propose standardized activities that may be ordered to obtain a complete formal process.

4.1 Constructs

IT AM process can be described as a set of phases and sub-phases, each one with a well defined purpose [2]. For space limitations we will just present part of the sub-phases (Table 3).

Table 3. IT AM Phases and Sub-Phases

Phases	Sub-Phases	Description	References
Planning	Establish audit objectives	Determination of what is intended to be accomplished with the audit according to the requirements analysis	[23][25][26]
	Establish audit scope and schedule	Scheduling of audit in cooperation with the audit entity	[2][8][23][26]
Preparation	Audit team selection	Selection of auditors to perform the audit	[23]
	Obtain preliminary background of audited areas	Performance of a preliminary survey of the area to be audited so as to understand what the audit will entail	[2][26]

The same analysis of the most known frameworks of the area as well as some of the most relevant literature was performed to elicit the main roles of IT AM process. The audit roles as well as the references from where we elicited them are in Table 4.

Table 4. IT AM Roles

Role	Reference
Audit Manager	[1][2][9][11][23][24][26][27]
Audit Team	[2][8][11][23][26]
Audited Entity	[1][8][11][24]

With a deeper analysis of the main literature and IT frameworks of the area, we identified what we consider to be the main activities for IT AM process, listed in Table 5 with the corresponding references. Since there are a high number of activities (54), here we just provide part of them.

Table 5. IT AM Activities and Responsibilities

Responsibility/ Activities	References
Periodically conduct internal audits to verify if everyone follows relevant guidelines for professional behavior, and process compliance	[22][28]
Obtain assurance of compliance and adherence to all internal policies derived from obligations	[22][27]
Audit must contribute to the improvement of risk management processes in the firm	[1][22][26]

4.2 IT Audit Management Process (BPMN)

After the definition of our constructs (IT audit phases, roles and activities), we are able to integrate our constructs and produce our model (IT AM process). We used the IT audit worldwide accepted phases (Figure 2) as the basis of our work.

To each phase we needed to analyze IT frameworks and main literature to elicit the sub-phases. Then, a similar procedure was done to elicit activities. Sub-phases have associated multiple activities and if we join the three in a hierarchical way, we have the basis of processes, sub-processes and tasks in the proposed IT AM Process.

Combining roles with the activities we can understand what each one do and combining activities with data we can understand which data is manipulated in each task and by whom. The integration was realized based on the context of each construct. Table 6 shows part of the relationships between our constructs.

After the constructs integration we are able to design our IT AM process. Due to space limitations, we only present one BPMN diagram as an example (Figure 3), which is the sub-process "Collection of Evidences and Issues".

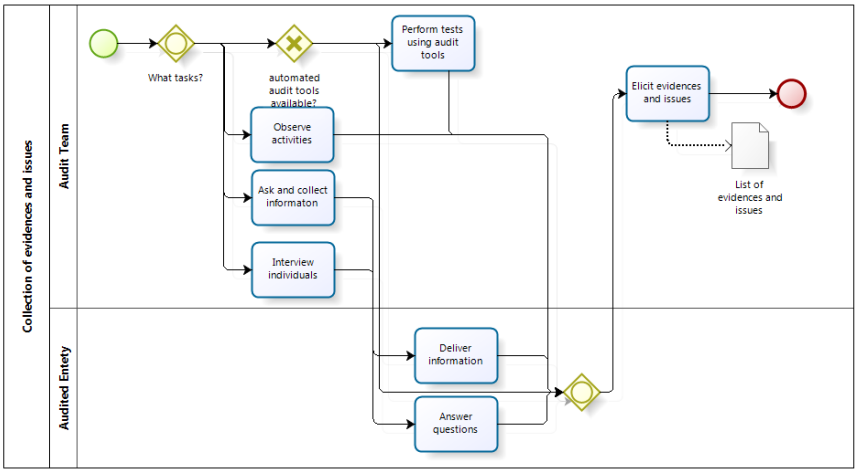


Fig. 3. Collection of Evidences and Issues

Table 6. IT AM Phases, Sub-Phases, Activities and Roles

Phases	Sub-phases	Responsibility/ Activities	Roles
Planning	Establish audit objectives	Plan and agree audit requirements	Audit Manager
		Write an audit plan, which must describe the objectives	Audit Manager
Preparation	Obtain preliminary background of audited areas	Gain preliminary understanding about the audited areas	Audit Team
		Perform documents and information assessment about relevant aspects of the audited entity	Audit Team
		Review the information relevant to audit assignments	Audit Team

4.3 IT Audit Management Information Architecture

Nowadays, organizations perceive the importance of linking business architecture to Information Architecture (IA) [29].This linkage, enable to manage the changes required by the business and maximize the benefits from the IT investments [29]. However, the current ad-hoc IA in place within many organizations cannot meet future needs because it has an incoherent framework, missing elements, few understood standards, low quality and unnecessary duplications [30]. Given such facts, we decided to design the IA of IT AM since it allows organizations to better manage their IT audit related information.

The entities represent business objects that can be seen as information or concepts that are necessary to support the business. Informational entities (IE) are the basis for modelling the IA since they represent information that is manipulated in processes. So, to provide a coherent IA we need to list all the entities elicited and provide a complete description. The entities are elicited from the constructs. Due to space limitations, Table 7 only shows part of the IE.

Table 7. Informational Entities

Entities	Identifier	Description
Objective	Objectives Description	Describe the objectives of an audit
Scope	Scope Description	Describe the scope of an audit such as physical locations, organizational units, activities and processes to be audited
Evidences	Type + Name	Represents all the information that can be used to prove some findings in an audit execution
Audit Report	Name + Date	Document that provides all the information about an audit (aggregates other information entities)

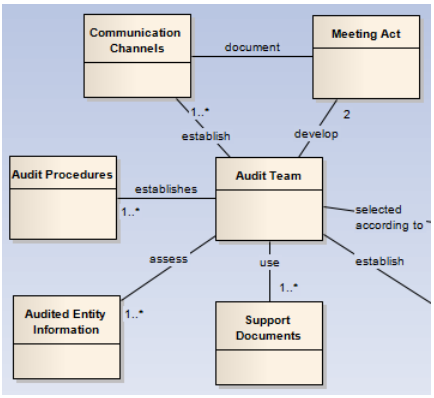


Fig. 4. Information Structure Viewpoint

The information structure viewpoint shows the structure of the information used in the organization or in a specific business process [31]. By space limitations Figure 4 only shows part of it.

4.4 IT Audit Management IS Architecture

Information systems architecture focuses on identifying and defining the applications and data considerations defining views that relate to information, knowledge, application services, and others. This research relates the information entities with the processes that form the IT AM process to elicit the applications needed to implement the process in an organization.

Firstly we provide a Create, Read, Update and Delete (CRUD) matrix (Figure 5) in order to identify clusters that represent application solutions. A CRUD matrix allows this because it is a communication model that represents communication interfaces among applications. With the CRUD matrix it is possible to understand the needed applications to perform the IT AM process, the information that each application manipulate, and the relations between applications. Secondly, to better visualize the cooperation between the various applications, we use the application cooperation viewpoint (Figure 6) and, finally, we use application structure viewpoint (Figure 7) to see the relation between applications and the information that is manipulated. Due to space limitations we only present part of the proposal in both Figures.

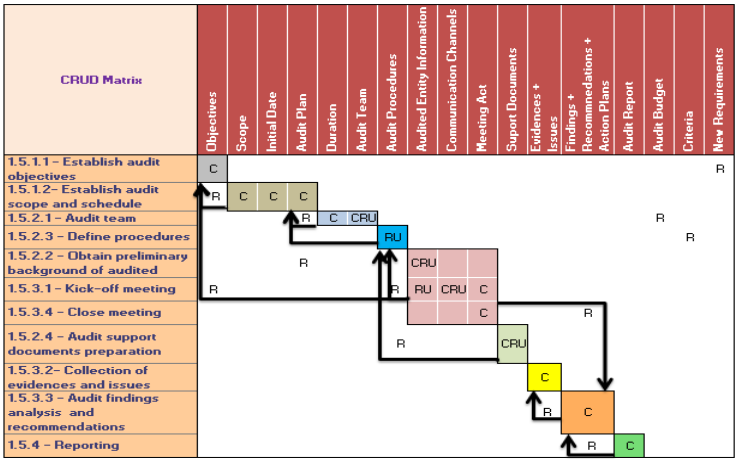


Fig. 5. CRUD Matrix

In the matrix we only represent the sub-processes that are composed by atomic tasks (we do not include any sub-processes). Due to our process decomposition, the other sub-processes do not have atomic tasks.

The application cooperation viewpoint shows the relation between application components. It describes the dependencies in terms of the information flows, or the

services they offer and use [31]. On the other hand, the application structure viewpoint shows the structure of one or more application components. It describes the structure of the applications through the sharing of information.

We can observe the usage of information common to the application components which provide a better representation of the relations between them.

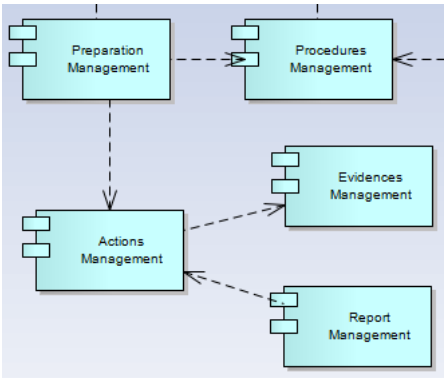


Fig. 6. Application Cooperation Viewpoint

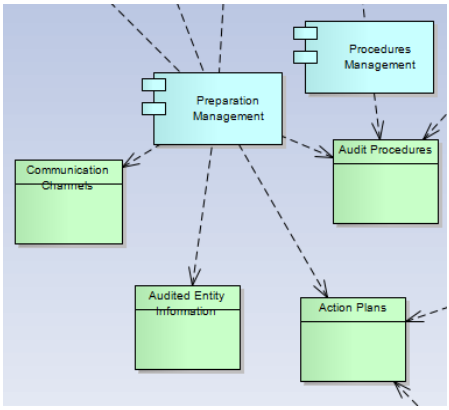


Fig. 7. Application Structure Viewpoint

5 Evaluation

After design our solution based on the main literature and IT frameworks of the area which gave us a strong theoretical viewpoint. To provide some practitioner viewpoint in order to include some industrial experience, we performed eight IT audit experts' interviews at Portuguese organizations.

In the interviews, we used open-response questions because of the nature of the information we needed to elicit. Furthermore, clarifications regarding the various concepts used by the respondents were sought during the conversation, so that later these descriptions could be examined and matched to more standard designations. The interviews were conducted over a one month period. Each session lasted from 30 to 60 minutes and was transcribed into digital data for analysis.

We used structured interviews, covering a diverse sample of organization types, sizes, and roles. Detailed information about the respondents is provided in Table 8. To support the interviews, we designed a questionnaire to support and lead the discussion.

The questionnaires aim at understanding if practitioners agree with the created models: IT AM process, IT AM information and IS architectures. The interviewees analyzed our work and classified it according to some factors (Table 9) provided by the data model quality framework of Moody and Shanks [32].

Table 8. Respondents Details

Id	Type	Area	Position	Work Experience
1	Telecom.	Information Systems	Director	Manager of Operations, Data Base Administration and Technical Support from 2002 to 2010; Sourcing and Staffing Manager since 2010
2	Consultant	IT Governance and Project Management	Senior Project Manager	SI Advisor from 1997 to 2001; Process Manager from 2001 to 2005; Practice Manager from 2009 to 2011 in the areas of IT Governance; Senior Manager from 2009 to 2011 in the areas of IT Governance;
3	Banking	Risk Management and IT Quality	Executive Administrator	Director in a IT Consulting firm from 1999 to 2000; Software Administrator at IT Services from 2000 to 2003< Administrator at an IT Consulting firm from 2003 to 2006 in the areas of SI Architecture, Risk Management and Processes and IT Quality
4	Banking	Standards and Operations	Executive Coordinator	Executive Coordinator from 1998 to 2012 in the area of Methodologies and Standards, Processes and Procedures, Organizational Good Practices, Control Department and Software Quality
5	Banking	Risk and Compliance	Director	Director at IT Risk and Compliance Department from 2007 to 2012
6	Banking	IT Management	Executive Manager	Software Development Manager from 2000 to 2005; IS Architectures Manager from 2006 to 2008; Project Office Manager from 2008 to 2010; IT Users Relationship and Logical Architecture Manager from 2010 to 1012
7	Consultant	IT Services Management	CEO	Quality Management in the implementation of systems from 1994 to 1997; Perform of audits in IT Infrastructures and Systems from 1997 to 2001; Design and Development of systems compliant with ISO 9001 from 1997 to 2001; Coordinator to Audit and Quality area at *Instituto de Informática* from *Ministério do Trabalho e da Solidariedade Social* (MTSS) from 2001 to 2011;
8	Consultant	IT Governance, EA and Enterprise Content Management	Business Practice Manager	Developer at a consulting firm from 2002 to 2003; Consultant at a firm from 2002 to 2006; Senior consultant at a firm from 2006 to 2008; Product manager from 2009 to 2010 in the area of modeling (BPM) and product quality; Project Manager from 2008 to 2011 in a consulting firm; Business Practice Manager since 2011 in a consulting firm

The questions goal is to verify if each factor was reached. Additionally, there was an open question in which respondents should provide a complementary commentary about our work. Each session lasted about 60 minutes and passed to digital data for analysis.

Next, we discuss each of the factors proposed in the Moody and Shanks framework and explain how our proposal reaches them. We also explain the changes made in our proposal in order to solve some problems that practitioners found. Given to space limitations we present part of the conclusions in Table 10 as an example.

Table 9. Moody and Shanks Factors

Factor	Description
Completeness	Completeness refers to whether the model contains all user requirements
Integrity	Integrity definition of business rules or constraints from the user requirements
Flexibility	Flexibility is defined as the ease with which the model can reflect changes in requirements without changing the model itself
Understandability	Understandability is defined as the ease with which the concepts and structures in the model can be understood
Correctness	Correctness is defined as whether the model conforms to the rules of the modeling technique (i.e. whether it is a valid model). This includes diagramming conventions, naming rules, definition rules, rules of composition and normalization
Simplicity	Simplicity means that the model contains the minimum possible entities and relationships
Integration	Integration is defined as the consistency of the model with the rest of the organization
Implementability	Implementability is defined as the ease with which the model can be implemented within the time, budget and technology constraints of the project

We describe the conclusions (column 2), the Moody & Shanks factor to which it refers to (column 3) and the analyzed model (column 4).Now we can discuss the conclusions presented in Table 10, as an example, to understand the kind of improvements performed (Table 11).

Table 10. Conclusions discussion

Nº	Conclusion
1	The first issue is solved through the definition of procedures in each audit. In the sub-process "Preparation" we have a task called "Define procedures" that intend to solve this problem. We have this more generic task that ensures an adaption of audit procedures according the type of audit. The second issue is pertinent but we don't solve it directly. In the sub-process "Reporting", we provide a task called "Distribute report" which, in spite of being more generic, indirectly guarantees that all the stakeholders receive the audit report.
2	Since we intend to provide a general and adaptable process, we cannot decompose the entity evidences. It is impossible to represent all the possible information that can be used as evidence, so, we maintain the entity evidence. This decision does not influence the quality of models since in the case of having multiple types of information, they have the same relations and purpose of the "Evidences" entity.
3	The access to applications of other domains is already visible in the CRUD matrix (Figure 9) when we have columns only with reads (R). We assume that the entity "Evidences" is created in this process because we need to save some information about it. The saved information can be just a link or a document's name.
4	As described in conclusion 1, first issue, we include mechanisms to support a sufficiently adaptable implementation such as the "Define procedures" task.
5	Already good form practitioners viewpoint.

Table 11. Practitioners Main Conclusions

Nº	Conclusion	Factor	Model
1	It is complete but to implement it, organizations need to complement some parts of the process according to the type of audit. For example, audits in the security domain need to complete it with specific procedures. The process does not clearly demonstrate that the Audit Report is delivered to the various stakeholders.	Compl.	IT AM Process
2	The information listed is sufficient to perform the audit. "Evidences" entity can be any type of information, so to represent them as a unique entity can be an abuse.	Compl.	IT AM Information Architecture
3	It can be necessary to access other applications that do not belong to the audit department. Also, some entities listed, such as "Evidences" entity, are usually collected using other systems, so it is necessary to be careful when it is said that evidences are created in the audit process.	Compl.	IT AM IS Architecture
4	From the audit stakeholder's point of view, the proposed process can be changed enough without losing integrity. It is important to have mechanisms to support an adaption of the process by organizations.	Integ.	IT AM Process
5	The information provided allows good integrity.	Integ.	IT AM Information Architecture

6 Conclusion

Since the evaluation was positive, we argue that the limitation pointed out by Goeken [33] was fulfilled (frameworks lack theoretical foundations). Plus, with the merging of the frameworks in IT AM activities, we argue that the limitation stated by Pereira and Mira da Silva [15] was fulfilled too (frameworks overlap each other). Finally, the limitations pointed by Rosário [10] were also solved since a complete IT AM process/architecture based on main IT frameworks and literature was achieved.

Our work aims to contribute to the IT AM process design, so that it is possible to have a formal way of performing audits. Knowing that the formalization of audit tasks is a difficult goal to achieve, we believe that our work is another step forward. The main contributes of this research are: the formalization of the IT audit adaptable to all types of organizations, based on both theoretical and practitioners' viewpoints; the design of a complete process where all the tasks, roles and data represented were collected from IT frameworks or by the most relevant literature. However, our work has some limitations too. A higher number of interviews should be performed to ensure more consistency and coherency as well as to study other types of organizations. Also, practitioner's functions and type of industry where they operate is limited.

In the future, this research can be completed with a more empirical work. Primarily we could observe at real organizations if their actual IT AM process/architecture is performed as designed here. If not, the observation of real audit activities could give us an idea of how ad-hoc conducted audits are and help understand what the differences to our process are as well as collect feedback. Then, to observe our work in real situations we could implement the proposed IT AM process and architecture in order to understand if this implementation is easy to make as our models evaluation seems to demonstrate or if there is any relevant constrain in the application of our proposal in real world organizations.

References

1. Tarantino, A.: Governance, Risk and Compliance Handbook: Technology, Finance, Environmental, and International Guidance and Best Practices. Wiley & Sons, Hoboken (2008)
2. Senft, S., Gallegos, F.: IT Control and Audit. Taylor & Francis Group, Boca Raton (2009)
3. Allen, D., Faff, R.: The Global Financial Crisis - some attributes and responses. Accounting and Finance 52, 1–7 (2012)
4. Mcdonough, A., Sackmann, S.: Compliance and Organization Value: How Markets React to Reported Lapses in Corporate Governance. In: Conference on Commerce and Enterprise Computing, pp. 239–244. IEEE Press, New York (2008)
5. Radovanovic, D., Radojevic, T., Lucix, D., Sarac, M.: IT audit in accordance with Cobit standard. In: 33rd International Convention on MIPR, pp. 1137–1141. IEEE, NY (2008)
6. Pai, P.F., Hsu, M.F., Wang, M.C.: Computer-Assisted Audit Techniques based on an Enhanced Rough Set Model. In: International Conference on Networked Computing and Advanced Information Management, pp. 207–212. IEEE Press, New York (2008)
7. Webster, J., Watson, R.T.: Analyzing the Past to Prepare for the Future: Writing a Literature Review. MISQ 26(2), xiii–xxiii (2002)
8. Grembergen, W.V., Haes, S.D.: Enterprise Governance of Information Technology: Achieving Strategic Alignment and Value. Springer, Heidelberg (2009)
9. Steinberg, R.M.: Governance, Risk Management, and Compliance: It Can't happen To Us - Avoiding Corporate Disaster While Driving Success. John Wiley & Sons, Hoboken (2011)
10. Rosário, T., Pereira, R., Mira da Silva, M.: Formalization of the Audit Process Management. Accepted to 15th EDOC Workshops. IEEE (2012)
11. De Haes, S., Grembergen, W.: Analysing the Relationship between IT Governance and Business/IT Alignment Maturity. In: 41st HCISS, p. 428. IEEE Press, New York (2008)
12. Decker, G., Barros, A.: Interaction Modeling using BPMN. In: International Conference on Business Process Management, pp. 208–219. ACM Press, New York (2007)
13. Griffin, P.A., Lont, D.H.: An Analysis of Audit Fees Following the Passage of Sarbanes-Oxley. Asia-Pacific Journal of Accounting & Economics 14, 161–192 (2007)
14. Pereira, R., Mira da Silva, M.: A Maturity Model for Implementing ITIL v3. In: 6th World Congress on Services, pp. 399–406. IEEE Press, New York (2008)
15. Pereira, R., Mira da Silva, M.: A Maturity Model for Implementing ITIL V3 in Practice. In: 15th IEEE International Enterprise Distributed Object Computing Conference Workshops, pp. 259–268. IEEE Press, New York (2008)
16. Sahibudin, S., Sharifi, M., Ayat, M.: Combining ITIL, COBIT and ISO/IEC 27002 in Order to Design a Comprehensive IT Framework in Organizations. In: 2nd Asia International Conference on Modeling & Simulation, pp. 749–753. IEEE Press, New York (2008)

17. Nicewicz-Modrzewska, D., Stolarski, P.: ITIL implementation roadmap based on process governance. In: European University of Information Systems, paper 124 (2008)
18. Osterle, H., Becker, J., Frank, U., Hess, T., Karagiannis, D., Krcmar, H., et al.: Memorandum on design-oriented information systems research. EJIS 20, 7–10 (2011)
19. Hevner, A.R., March, S.T.: Design Science in Information Systems Research. MISQ 28(1), 75–105 (2004)
20. March, S.T., Smith, G.F.: Design and natural science research on information technology. Decision Support Systems 15, 251–266 (1995)
21. Pereira, R., Mira da Silva, M.: Towards an Integrated IT Governance and IT Management Framework. Accepted to 16th EDOC. IEEE (2012)
22. The Institute of Internal Auditors: International Standards For The Professional Practice of Internal Auditing (2010), https://na.theiia.org
23. International Standard Office: ISO 19011 - Guidelines for quality and/or environmental management systems auditing. Geneva (2002)
24. Thomson Reuters: Fundamental of GRC: The Connected Roles of Internal Audit & Compliance. White Paper (2011)
25. Carlin, A., Gallegos, F.: IT Audit: A Critical Business Process. Computer 40(7), 87–89 (2007)
26. Davis, C., Schiller, M., Wheler, K.: IT Auditing: Using Controls to Protect Information Assets. McGrawHil, New York (2011)
27. Information Technology Governance Institute: IT Governance Institute: COBIT 4,1 (2007), http://www.isaca.org
28. International Standard Office: ISO/IEC 38500 - Corporate governance of information technology. Geneva (2008)
29. Kamath, S.: Capabilities and Features: Linking Business and Aplication Architecture. In: Conference on Software Architecture, pp. 12–21. IEEE Press, New York (2008)
30. Watson, R.W.: An enterprise information architecture- a case study for decentralized organizations. In: 33rd HCISS, pp. 1–10. IEEE Press, New York (2008)
31. Lankhorst, M., et al.: Enterprise Architecture at Work - Modelling, Communication and Analysis. Springer, Heidelberg (2009)
32. Moody, D.L., Shanks, G.G.: Improving the quality of data models empirical validation of a quality management framework. Information Systems 28(6), 619–650 (2003)
33. Goeken, M., Alter, S.: Towards Conceptual Metamodeling of IT Governance Frameworks Approach – Use – Benefits. In: 42nd HCISS, pp. 1–10. IEEE Press, New York (2008)

Towards an Architecture for Collaborative Cross–Organizational Security Requirements Management

Christian Sillaber, Michael Brunner, and Ruth Breu

University of Innsbruck, Department of Computer Science, 6020 Innsbruck
{christian.sillaber,mike.brunner,ruth.breu}@uibk.ac.at

Abstract. Organizations increasingly adopt or consider adopting external services hoping for higher flexibility and reduced costs. However, currently existing deficiencies of processes and tools force service consumers to renounce from the expected advantages and to trade off profitability against security. These security and compliance concerns are predominately due to negligence or manual resolution of security policy and configuration dependencies, caused by distinct terminologies, languages and tools used at both the service provider and service customer. To overcome these kind of problems in the collaborative cross–organizational security management, we have developed CoSeRMaS, a collaborative and semi–automated tool to manage, define and validate inter- and cross–organizational security requirements. This paper introduces the CoSeRMaS prototype and gives an overview of the features that have been developed.

Keywords: Collaborative Security Requirements Management, Business Security Requirements, Change-driven Security, Living Security, Workflow-driven Security Requirements Engineering, Security Requirements Meta-model.

1 Introduction

Driven by the need to save costs and to allow for higher flexibility, organizations increasingly adopt or consider adopting services from service providers to support their internal and external workflows. The use of external services from service providers allows organizations to focus on their core business [1] and enables the rapid development of new business models that build solely on the dynamic composition of external services [2]. The evident resulting dependency on the service provider requires the service provider to achieve and maintain compliance with security requirements so that service customers can trust them.

The research domain *cross–organizational security management* is relatively new. While much literature and business initiatives for organization-internal security management exist, cross–organizational security management, where several organizations are involved in achieving and maintaining security requirements, is neglected in both research and practice [3]. The increasing complexity

W. Abramowicz (Ed.): BIS 2013, LNBIP 157, pp. 199–210, 2013.

of security requirements that demand coordinated activity of various service organizations has been identified as a major obstacle for the adoption of cloud services in organizations [3, 4]. Recent research has shown that organizations, seeking to adopt external services, face various challenges across all layers of the cross–organizational landscape. On the IT layer, organizations are faced with non-standardized interfaces [5,6] and a multitude of architectural guidelines and frameworks [7]. On the business layer, most organizations see availability of service [8, 9], performance issues [3, 7, 10] and proper scalability [3, 10] as major obstacles. Trust and liability issues [3,9,11] as well as the definition and enforcement of service level agreements [12, 13] and a multitude of legal and regulatory rules that must be complied with are the most frequently identified challenges on the management layer [3, 11].

These challenges require thorough and rigorous management and coordination across all layers of the cross–organizational service landscape. While different solutions have been proposed in the past that address some of these challenges, they almost always limit themselves to a narrow set of problems (e.g. Cloud Service reference models [14–17] that neglect the governance aspect) or do not take the specifics of cross–organizational security management into account [3]. Although organizations would benefit heavily from a framework and tool-set that integrates all layers necessary for cross–organizational security requirement management, to the best of our knowledge no such tool has yet been presented. To address this gap in research, we present CoSeRMaS, the **Co**llaborative **Se**curity **R**equirements **Ma**nagement **S**ystem (`http://cosermas.q-e.at`). To help organizations manage complex cross–organizational security requirements, CoSeRMaS provides an innovative framework which can handle the systematic documentation, analysis, reporting and management of security requirements across cross-organizational layers.

The remainder of this paper is structured as follows: An overview on related work is given in Section 2. Section 3 introduces the architecture of CoSeRMaS. In Section 4 the application itself is presented and the paper is concluded in Section 5.

2 Related Work

Cross–organizational security requirements management can be seen as a tool in the context of corporate governance, security requirements engineering, compliance management and audit management. This section gives an overview on existing work in this area and highlights the existing gaps that our approach addresses.

In [11, 18, 19] governance models for cloud service providers are introduced. The presented frameworks focus on the business and managerial layers of cloud service organizations and fail to address the important aspect of establishing links between all model elements across all layers. Also, we found that they lack a reference implementation and broad tool support.

In [20], the authors present a requirement based access control analysis and policy specification method. The presented method integrates access control analysis to ensure a policy and requirements compliant system. A set of process descriptions and heuristics are presented that support analysts to derive and specify access control policies while ensuring traceability. However, the presented approach focuses on software development processes and not on general business processes or cross–organization service compliance. The CoSeRMaS approach closes this gap by providing a cross–organizational compliance and security requirements management system.

Several methods have been proposed in the past to support cross–organizational security requirements management by formalizing laws (most commonly privacy laws). Most approaches (e.g. [21,22]) employ first-order logic models to derive legislative objectives. While these approaches work well for automated systems and processes, they lack support for higher level business processes. I.e. they can be used to validate a specific software tool for its compliance but do not provide adequate means beyond that on an organizational level. Furthermore, they do not provide traceability mechanisms and are often not designed for use by untrained stakeholders.

Several Governance, Risk and Compliance (GRC) tools are currently available on the market that provide support for organization wide compliance and risk management [23]. The objective of GRC tools is to simplify, systematize, centralize, and automate key controls. Furthermore, they are used to manage processes, systematize and centralize documentation, and monitoring. Almost all GRC tools provide dashboard like features supporting the definition of key risk indicators and the continuous monitoring of controls. While the provided dashboards provide various views, they often fail to present the link between concepts on the higher and lower layers in a clear and concise matter as discussed before. While GRC tools like Axentis [24], BWise [25] or OpenPages [26] provide support for risk and compliance management, their inner processes are often not publicized and their scientific validity is not verifiable [27]. It is often unclear how well these tools support the functional model of the company or to what degree they require the company to align their business processes with a specific methodology [27,28]. A recent study [7] indicates that currently existing GRC tools do not provide enough flexibility or expressiveness to support the organization's infrastructure to a satisfactory degree. Also, most GRC tools use a fixed functional model that is often hard to adapt to the specifics needs of an organization. These deficits are addressed by the highly expressive asset and protection target model that will be presented in the remainder of this paper.

3 Architecture

The Security Requirements meta-model is depicted in Figure 1. It builds on the concepts introduced in the Business Security Meta Model [29] and incorporates the Living Security paradigm [30,31] to address functional and non-functional requirements for corporate governance, security requirements engineering and

compliance management [23, 29]. In essence, these building blocks provide a process model for evolving security critical systems. It merges concepts from IT management and security engineering to conceive a collaborative approach for the continuous engineering and management of security requirements. The central components of the model (a refined sub-set of the Business Security Meta Model), are the *Security Requirements* and the *Protection Targets*.

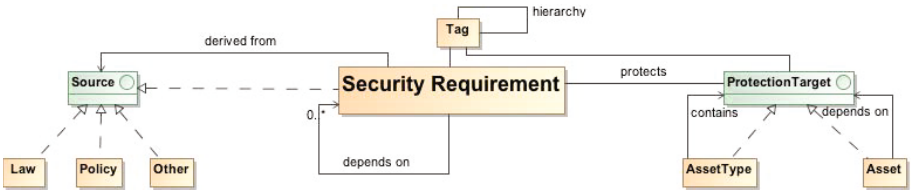


Fig. 1. Meta-model used by CoSeRMaS

Security Requirements can be composed of subordinate–requirements through the *realizes* relation allowing to build an arbitrary requirement fulfillment tree Each security requirement may be derived from a *Source* being either another security requirement, a law, a policy or some other relevant document such as a service level agreement.

While it seems superficial to include both the *derived-from* and *realizes* relation in the model as they seemingly describe the same concept, this distinction serves an important purpose as many security requirements, especially at the IT layer, often depend on other security requirements that are not derived from them. One simple example that we often came across during the evaluation that demonstrates the issue: The security requirement "ensure availability of payment processing service" (derived from a policy) depends, among others, on the security requirement "provide timely backups for payment processing data" which itself depends on security requirements that are derived from various customer specific service level agreements. Therefore, by using both relations a very expressive model can be created that not only considers hierarchical dependencies between security requirements, but also enables expressive connections linking the content of security requirements together. To summarize, the semantics between those two types of connections differs in that the *derived-from* relation serves primarily documentation purposes while the *realizes* relation models direct dependencies that propagate changes within the security requirement model.

How the therein described Security Requirements interact internally and with external organizational interfaces, the CoSeRMaS meta-model and the fulfillment-model (describing the state of the Security Requirements) are described in the remainder of this section. To make the meta-model usable in an organizational context, a defined set of roles is necessary. To ensure adequate levels of trust in collaborative security engineering processes, a stakeholder model was developed that defines roles and their scope of responsibility. The remainder of this section

introduces said stakeholder model and presents the meta–model as well as the security requirement fulfillment model in more detail.

3.1 The Stakeholder Model

Before introducing the meta-model in more detail, this section presents the stakeholder model that was developed for CoSeRMaS. It defines the existing roles responsible for managing the Security Requirements (cf. fig. 2).

Fig. 2. The stakeholder model

Every security requirement is related to one or more stakeholders. The relationship between security requirements and stakeholders describes certain types of responsibilities and allows to consider parties that are interested in specific requirements such as customers or business partners. In CoSeRMaS, three types of stakeholders exist that are actually dealing with security requirements, their definition, refinement and their fulfillment status. These are the *basic users*, *requirement engineers* and *chief requirement engineers*. Basic users are responsible for confirming or revoking the fulfillment status of security requirements that have been assigned to them. To ensure the integrity of the security requirements model, they may only refine requirements within their scope of responsibility by requesting the appropriate modification (i.e. creation of subordinate requirements) from the *requirement engineers*. Requirement engineers are managing the security requirements model by creating new or altering the existing security requirements. Furthermore they define and refine the dependencies among security requirements. To allow for scalability and to provide support for the corporate structure of organizations (e.g. separation concern and to model organizational hierarchies), a small number of *chief requirement engineers* oversees and manages groups of subordinate requirement engineers and the development of the complete security requirement model.

Auditors can be provided with access to reports and log data (read only). Access to the actual security requirement model is limited to a predefined subset (i.e. audit scoping). The responsibility of *Administrators* focuses on typical administrative tasks as user management, backup management and managing to link to external tools that provide the landscape model. The stakeholder model is kept very simple yet expressive enough to sufficiently model all aspects of corporate structure [29].

3.2 The CoSeRMaS Meta–model

After introducing the CoSeRMaS stakeholder model, this section presents the security requirements management meta–model in more detail. The design rationale

for the meta–model and evaluation of already existing approaches are extensively discussed in [29, 32, 33].

Security Requirements are defined by title, description and links to super-ordinate and sub-ordinate security requirements. Furthermore, the *revalidation model* that is assigned to each requirement specifies after which period of time the fulfillment status of a requirement automatically expires (other fulfillment mechanisms can be freely defined through OCL). Each requirement is linked via the *responsibleUser* relation to the basic user or (chief) requirements engineer that confirms or revokes the requirement's fulfillment status.

The state of each requirement can take any of the following six values: Newly created requirements take the *Added* state. Those security requirements that are in need of further refinement (e.g. as requested by a basic user) take the *Refinement* state. Security requirements that are deleted are not actually removed, but take the *Deleted* state for documentation purposes. Also, they can be restored by chief requirement engineers. The remaining three states *Fulfilled*, *Partially fulfilled* and *Not fulfilled* denote whether the requirement and its sub-ordinate requirement matches the reality it describes. These states and the formal relationship between them (the fulfillment model) is explained in more detail in the following section.

The link between the higher level security requirements and the service and IT layer is established via *Protection targets*. The generic meta–model can be used to describe any functional model (i.e. business processes, IT infrastructure elements, etc...) within organizations. These functional model elements are directly linked to specific security requirements. This is achieved through the *protects* relation. Protection targets either correspond to abstract functional model elements (AssetTypes, e.g. "database server") or specific instances (Asset, e.g. "database server with id xyz") and can be arranged in a hierarchical manner.

Besides their main purpose of modeling abstract functional model elements, *AssetTypes* can also be used to group assets together. The example shown in Figure 3 shows the AssetType *External IT Services*. This AssetType contains three assets: *Service 1*, *Service 2* and *Service 3*. Through additional attributes (i.e. tags), the assets can be further described. In the example the attribute *BDSG-Sensitive-Data* = {*true* / *false*} is assigned to external services to mark those that process sensitive data[1]. This allows for novel security requirement rules as: *"Whenever an asset where the attribute BDSG-Sensitive is true is created, a trigger rule ensures that only BDSG compliant assets connect to it"*. As the meta–model defines the protection targets in a very generic way, any functional model is supported and the organization does not have to adhere to a specific paradigm or framework. Confer to [29] for more detailed information on the security meta-model and the connection to the functional model elements.

The manual *Confirmation* and *Revocation* of a specific security requirement can be done by the responsible basic user. When making the change the user can

[1] German Federal Privacy law. [34] demonstrated an new method for organizations to manage and ensure compliance of model elements on the IT layer with that privacy law in cross–organizational environments.

also attach additional information for justifying and further improving the documentation of the change. **Key-Value attributes**, as already outlined in the previous example (e.g. the boolean value of whether a specific service requires special privacy security requirements as shown in Figure 3) bring structured storage of additional information to security requirement and protection targets. These attributes provide organizations with enough freedom to incorporate aspects from other domains such as risk management into the definition of security requirements.

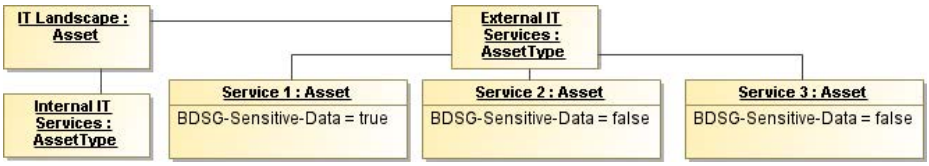


Fig. 3. Exemplary Protection Targets from the case study presented in [34] that model an IT Service Landscape with three external IT services

To express the various forms of requirements, CoSeRMaS introduces the following four types of security requirements. **Basic Requirements** are the main building blocks for most instances. They provide means to build arbitrary requirement dependency trees. They represent static requirements where the higher-level requirements are managed by (chief) requirement engineers and the lower-level requirements are managed by the responsible basic user. Basic requirements can propagate their fulfillment status to super-ordinate requirements. **Message-based Requirements** support requirement confirmation through structured communication channels such as e-mail, web-access or a RESTful web service with external users. The benefit from this requirement type is that responsible stakeholders do not need direct access to the system to confirm their requirements. Obviously, it is not possible to define sub-requirements for message-based requirements as message-based requirements are leaf[2] requirements. **Auto-check Requirements** allow the integration of external tools, such as reactive password checkers, port scanners or access control verification tools to automatically check the requirements fulfillment status. For obvious reasons, this requirement type also does not allow sub-ordinate requirements. **Rule-based Requirements** offer means to model complex requirements and their behavior. Each requirement contains one or more action rules which are either triggered by changes within other requirements or assets (e.g. new server added, server moved from one legislative zone to another one, update of application software updated requirement description, changed policy document, etc.). These actions range from the creation/removal of requirements to the automatic confirmation as well as the revocation of confirmations.

[2] The term "leaf requirement" stems from the fact that requirements build trees and thus leafs refer to requirements without further subordinate requirements.

3.3 Fulfillment Models

This section introduces the fulfillment model that is used to formally describe the fulfillment dependencies between security requirements.

Let r be a security requirement. The fulfillment state $FS(r)$ of a security requirement r is either `FULFILLED` (F) meaning the security requirement matches the reality it describes (e.g. the security requirement states that the server software must be up-to-date and the server software is in fact up-to-date). Or the state is `PARTIALLY_FULFILLED` (P) meaning that the attributes or aspects described by the security requirement do not match with reality for 100% - e.g. two out of three subordinate requirements are `FULFILLED`. The third state a security requirement can have is `NOT_FULFILLED (N)` which means that the security requirement does not match the reality at all. We assign the value 1 to every requirement meeting the condition F, 0 to every requirement meeting the condition N and a value $\{n|n \in \mathbb{R} \wedge 0 < n < 1\}$ to requirements that are partially fulfilled. The fulfillment state of those security requirements that do not depend on the state of subordinate requirements is set directly by the user or an automated script:

$$FS(r) = \begin{cases} 1 & \texttt{FULFILLED} \\ 0 & \texttt{NOT\_FULFILLED} \end{cases}$$

For a requirement r', whose state is the result of the evaluation model $FM_{r'}$, applied to the state of its subordinate requirements $(r_1, r_2, \ldots, r_n)$, we define the fulfillment state as follows: $FS(r', FM_{r'}) = FM_{r'}(FS(r_1, FM_{r_1}), FS(r_2, FM_{r_2}), \ldots, FS(r_n, FM_{r_n}))$ and $FM_{r'} : FS(r_1, FM_{r_1}) \times FS(r_2, FM_{r_2}) \times \ldots \times FS(r_n, FM_{r_n}) \rightarrow \{n|n \in \mathbb{R} \wedge 0 \leq n \leq 1\}$ We define the basic fulfillment model FM_{basic} for security requirements r without subordinate requirements as a mapping to their respective fulfillment state $FS(r) \in (0, 1)$. For convenience, we define the following equivalence: $FS(r) \equiv FS(r, FM_{basic})$ This recursive, arithmetic definition allows for the convenient definition of the three - most often needed - fulfillment models besides the basic fulfillment model:

Let FM_{and} be the fulfillment model that evaluates to 1 if the state of all subordinate requirements (and their subordinate requirements) evaluates to 1. If not all subordinate requirements evaluate to 1, FM_{and} evaluates to 0. Let FM_{or} be the fulfillment model that evaluates to 1 iff at least one subordinate requirement evaluates to 1; 0 otherwise. Let $FM_{threshold}(x)$ be the fulfillment model that evaluates to 1 iff the average fulfillment state of all subordinate requirements is equal or greater than x. If this threshold is not met, it evaluates to 0. While most cases can be properly covered by combining the previously described fulfillment models, corner cases and special requirements can be easily met by creating custom fulfillment models. The described model has already been used to successfully formalize a set of security requirements derived from the German Federal Privacy Law [34] as well as specific business security requirements from a large multinational B2B network provider [32].

4 The CoSeRMaS Application

To implement the concepts presented in the previous section, CoSeRMaS [29] has been developed as a novel tool to collaboratively manage security requirements. Since the tool contains a large set of features and many different report views, it is impossible to present them all in detail. Therefore we only showcase a small subset of the tool's core functionality in this section.

After logging in to the CoSeRMaS web application, from any web browser, the stakeholder is presented with a welcome screen, that gives him a short overview on the security requirements assigned to him and their respective fulfillment status in the form of both a list and a pie chart. Various *Requirements Views*, provide the stakeholder with lists of all requirements he is responsible for. These tabular view show the title, the fulfillment status, confirmation date as well as additional information for each requirement.

Figure 4 shows the graph view for security requirements provided by CoSeRMaS. In this view, the requirements and their dependencies are shown as an interactive graph. This graph can be navigated by the stakeholder. The graph itself can also be changed from the general view (cf. Figure 4) — where each requirement is seen within the entire requirement graph to the contextual view. In the contextual view that is presented in Figure 5, a selected requirement is shown with more details, including its relationship to the source it is derived from. Furthermore, the subordinate requirements as well as the link to the relevant protection targets is graphically shown and can be interactively navigated.

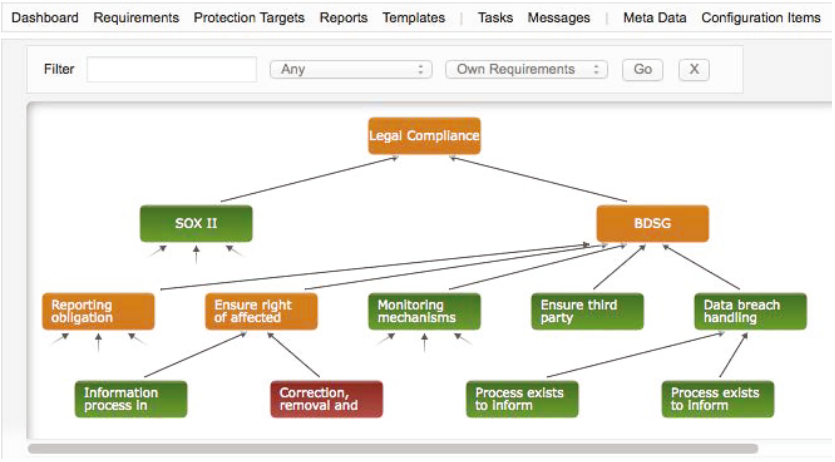


Fig. 4. Interactive graph view of all security requirements assigned to the stakeholder

In accordance with the stakeholder model, requirement and chief requirement engineers can create new requirements or refine existing ones (i.e. creating subordinate requirements) from within CoSeRMaS. During the creation of new

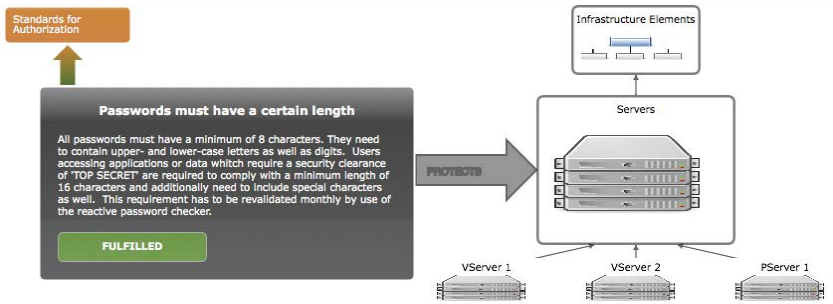


Fig. 5. This view shows a security requirement that is linked to an "abstract" server. All instances of this server are then automatically linked to the security requirement

requirements, the link between the new requirement and corresponding protection targets (i.e. elements of the IT or Business Process landscape) can be established. From an **end-user perspective** the tool provides an extensible and distributed approach to the management of security requirements. As there are several organization and user specific functional and non-functional requirements to be addressed, two large scale user studies with our industry partners in the domains of digital rights management and health care management are currently in preparation.

5 Conclusion

CoSeRMaS is a novel tool that integrates the functionality required for the collaborative cross–organizational management of security requirements. It provides a highly expressive model for both the security requirements and the IT and process landscape of the organization. Through innovative means of linking the model elements across all organizational layers, it is possible to manage the entire set of security requirements on both a very abstract and – if needed at a very detailed level. The development of CoSeRMaS has been aligned with KPIs from project partners that were well met and a large scale evaluation with key stakeholders is also in preparation. Experiences with the formalization of laws and security policies from a large corporation show that the change driven process approach to security requirements engineering is rather straight forward.

Acknowledgments. This work was partially funded by the European Commission under the FP7 project "PoSecCo" (IST 257129) and supported by the project *QE LaB – Living Models for Open Systems (FFG 822740)* and was also partially conducted within the competence network Softnet Austria II and funded by the Austrian Federal Ministry of Economy, Family and Youth, the province of Styria, the Steirische Wirtschaftsfoerderungsgesellschaft mbH and the city of Vienna in terms of the center for innovation and technology.

References

1. Motahari-Nezhad, H.: Outsourcing business to cloud computing services: Opportunities and challenges. In: 2010 4th IEEE International Conference on Digital Ecosystems and Technologies (DEST), vol. 4, pp. 91–112 (2010)
2. Weinhardt, C., Anandasivam, A., Blau, B., Borissov, N., Meinl, T., Michalk, W., Stößer, J.: Cloud Computing – A Classification, Business Models, and Research Directions. Business & Information Systems Engineering 1(5), 391–399 (2009)
3. Thalmann, S., Bachlechner, D., Demetz, L., Maier, R.: Challenges in Cross-Organizational Security Management. In: 2012 45th Hawaii International Conference on System Science (HICSS), pp. 5480–5489 (2012)
4. Kandukuri, B.R., Ramakrishna Paturi, V., Rakshit, A.: Cloud security issues. In: Proceedings of the 2009 IEEE International Conference on Services Computing, SCC 2009, pp. 517–520. IEEE Computer Society, Washington, DC (2009)
5. Hofmann, P., Woods, D.: Cloud computing: the limits of public clouds for business applications. IEEE Internet Computing 14(6), 90–93 (2010)
6. Takabi, H., Joshi, J., Ahn, G.: Security and privacy challenges in cloud computing environments. IEEE Security & Privacy 8, 25–31 (2010)
7. Racz, N., Panitz, J., Amberg, M.: Governance, risk & compliance (grc) status quo and software use: Results from a survey among large enterprises. In: Proceedings of the Australasian Conference on Information Systems, vol. (21), pp. 337–347 (2010)
8. Kantarcioglu, M., Bensoussan, A., (Celine) Hoe, S.R.: Impact of security risks on cloud computing adoption. In: 2011 49th Annual Allerton Conference on Communication, Control, and Computing (Allerton), vol. 49, pp. 670–674 (2011)
9. Shaikh, F., Haider, S.: Security threats in cloud computing. In: Proceedings of the 2011 International Conference for Internet Technology and Secured Transactions (ICITST), pp. 11–14 (December 2011)
10. Jing, X., Jian-Jun, Z.: A Brief Survey on the Security Model of Cloud Computing. In: 2010 Ninth International Symposium on Distributed Computing and Applications to Business, Engineering and Science, vol. 9, pp. 475–478 (2010)
11. Guo, Z., Song, M., Song, J.: A Governance Model for Cloud Computing. In: Proceedings of the 2010 International Conference on Management and Service Science (MASS), vol. (2007) (2010)
12. Alhamad, M., Dillon, T., Chang, E.: Service Level Agreement for Distributed Services: A Review. In: 2011 IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing, vol. 9, pp. 1051–1054 (2011)
13. Wang, M., Wu, X., Zhang, W., Ding, F., Zhou, J., Pei, G.: A Conceptual Platform of SLA in Cloud Computing. In: 2011 IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing, vol. 9, pp. 1131–1135 (2011)
14. Nurmi, D., Wolski, R., Grzegorczyk, C., Obertelli, G., Soman, S., Youseff, L., Zagorodnov, D.: The eucalyptus open-source cloud-computing system. In: 9th IEEE/ACM International Symposium on Cluster Computing and the Grid, CCGRID 2009, pp. 124–131. IEEE (2009)
15. Santos, N., Gummadi, K.P., Rodrigues, R.: Towards trusted cloud computing. In: Proceedings of the 2009 Conference on Hot topics in Cloud Computing, p. 3. USENIX Association (2009)
16. Lenk, A., Klems, M., Nimis, J., Tai, S., Sandholm, T.: What's inside the Cloud? An architectural map of the Cloud landscape. In: Proceedings of the 2009 ICSE Workshop on Software Engineering Challenges of Cloud Computing, pp. 23–31 (2009)

17. Zhang, L.J., Zhou, Q.: CCOA: Cloud computing open architecture. In: IEEE International Conference on Web Services, pp. 607–616. IEEE (2009)
18. Sedaghat, M., Hernandez, F., Elmroth, E.: Unifying Cloud Management: Towards Overall Governance of Business Level Objectives. In: 2011 11th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, vol. 11, pp. 591–597 (2011)
19. Ahmad, R., Janczewski, L.: Governance Life Cycle Framework for Managing Security in Public Cloud: From User Perspective. In: 2011 IEEE 4th International Conference on Cloud Computing, vol. 4, pp. 372–379 (2011)
20. He, Q., Otto, P., Antón, A.I., Jone, L.: Ensuring compliance between policies, requirements and software design: A case study. In: Fourth IEEE International Workshop on Information Assuranc, vol. 4, pp. 209–221 (2006)
21. Basin, D., Klaedtke, F., Müller, S.: Monitoring Security Policies with Metric First-order Temporal Logic. Control 12, 23–33 (2010)
22. Lam, P.E., Mitchell, J.C., Sundaram, S.: A formalization of HIPAA for a medical messaging system. In: Fischer-Hübner, S., Lambrinoudakis, C., Pernul, G. (eds.) TrustBus 2009. LNCS, vol. 5695, pp. 73–85. Springer, Heidelberg (2009)
23. Tarantino, A.: Governance, Risk, and Compliance Handbook: Technology, Finance, Environmental, and International Guidance and Best Practices. Wiley (2008)
24. Bagranoff, N.A., Henry, L.: Choosing and Using Sarbanes-Oxley Software. Information Systems Control Journal 2, 49–51 (2005)
25. Spies, M.: A software assurance evidence approach to cloud security. In: 2011 22nd International Workshop on DEXA, pp. 39–43. IEEE (2011)
26. Racz, N., Weippl, E., Bonazzi, R.: IT Governance, Risk & Compliance (GRC) Status Quo and Integration: An Explorative Industry Case Study. In: 2011 IEEE World Congress on Services (SERVICES), pp. 429–436. IEEE (2011)
27. Racz, N., Weippl, E., Seufert, A.: Governance, Risk & Compliance (GRC) Software-An Exploratory Study of Software Vendor and Market Research Perspectives. In: 2011 44th Hawaii International Conference on System Science, pp. 1–10. IEEE (2011)
28. Sadiq, S., Governatori, G.: A methodol. In: Handbook of Business Process Management. Springer (2009)
29. Breu, R., Farwick, M., Innerhofer-Oberperfler, F., Brunner, M., Julisch, K., Karjoth, G.: D2.1 A Framework for Business Level Policies. Technical Report 257129, PoSecCo project (project no 257129) FP7 (2011)
30. Innerhofer-Oberperfler, F., Hafner, M., Breu, R.: Living Security - Collaborative Security Management in a Changing World. Parallel and Distributed Computing and Networks/720: Software Engineering 23, 467–489 (2011)
31. Breu, R.: Ten Principles for Living Models - A Manifesto of Change-Driven Software Engineering. In: 2010 International Conference on Complex, Intelligent and Software Intensive Systems, vol. 12, pp. 1–8 (2010)
32. Sillaber, C., Kalb, P., Breu, R.: D2.3 Software for a model-driven policy design. Technical report, PoSecCo project (project no 257129), 7th Framework Programme for R&D (FP7) (2012)
33. Brunner, M.: Specification and architecture for a tool to manage business security requirements based on enterprise architecture management. Master's thesis, University of Innsbruck, Austria (2013)
34. Sillaber, C., Breu, R.: Managing legal compliance through security requirements across service provider chains: A case study on the German Federal Data Protection Act. In: Informatik 2012: Proceedings der GI/GMDS-Jahrestagung, pp. 1306–1317. Gesellschaft fuer Informatik (GI) (2012)

Conceptual Architecture of Knowledge Base for Administrative Procedure Execution

Sergiusz Strykowski and Rafał Wojciechowski

Poznań University of Economics
Mansfelda 4, 60-854 Poznań, Poland
{s.strykowski,r.wojciechowski}@ue.poznan.pl

Abstract. The majority of public services require public offices to perform some formal administrative procedures. An increasing number of offices develop models of these procedures. The more detailed model the better because the quality of public office work depends less on the knowledge and competences of a specific employee. However, such detailed models would overwhelm an employee's list of to-do tasks with a great number of minor, routine activities. The remedy to this problem is automation of those tasks in an IT system. However, to make this happen, the system must be provided with the knowledge of all relevant circumstances affecting the courses of administrative procedures and manners of performing tasks included in these procedures. In this paper, a conceptual architecture of Administrative Knowledge Base for administrative procedure execution is proposed. The knowledge base assumes collecting three types of knowledge: common legal knowledge, public office knowledge, and administrative procedure knowledge, which are necessary for the automated execution of routine tasks.

Keywords: e-government, administrative procedures, knowledge base.

1 Introduction

The importance of information and communication technologies (ICTs) in everyday life of citizens and businesses is the cause of rising customer expectations for fast and effective public services. A step towards this is the use of ICTs, and particularly the Internet, as a tool to achieve better government, what is called e-government [1].

The majority of public services require a public office to perform some formal administrative procedures. An administrative procedure consists of a sequence of tasks aimed at resolving a case of a citizen or business through an administrative decision. Examples are: a procedure for issuing a building permit or a procedure for issuing a permit to cut trees and shrubs. Administrative procedures performed by public offices can thus be seen as the equivalent of core business processes in companies. However, just as companies, public offices also execute supporting business processes, such as hiring and releasing employees, or handling requests for days-off. Modeling administrative procedures is conceptually similar to modeling business processes. The main difference is about the primary source of guidelines on their form—in business processes it is usually operational practices while in administrative procedures it is provisions of law.

W. Abramowicz (Ed.): BIS 2013, LNBIP 157, pp. 211–222, 2013.

In most cases, models of administrative procedures are developed quite generally—mainly to ensure a standard way of execution, regardless of the competence and knowledge of a clerk carrying it out. The main advantage of the general approach is that the models are correct, despite changes in the law. The main disadvantage—during execution it is necessary for a clerk to interpret the general statements in models and translate them into appropriate operational activities to be performed [2]. The quality of the work depends on the competence of individual clerks; there is also an increased likelihood of errors.

It is possible to try to develop models in a more detailed form. In this case, each general task must be decomposed into several detailed ones. However, in the runtime phase, a clerk's list of to-do tasks would be then overwhelmed by a large number of minor tasks, mainly consisting in entering into a workflow system single data on the various aspects of procedures being carried out.

The remedy to this problem is automation of those tasks with an IT system. However, to make this happen, the system must be provided with the knowledge of all relevant circumstances affecting the courses of administrative procedures and manners of performing tasks included in these procedures.

In this paper, a conceptual architecture of Administrative Knowledge Base for administrative procedure execution is proposed. The knowledge base assumes collecting three types of knowledge: common legal knowledge, public office knowledge, and administrative procedure knowledge, which are necessary for the automated execution of routine tasks.

The remainder of this paper is organized as follows. Section 2 provides an overview of related work in knowledge-based modeling of administrative procedures. In Section 3, the proposed conceptual architecture of the Administrative Knowledge Base is presented. Section 4 presents the application of the knowledge base with an example of an administrative procedure for granting a building permit. Finally, Section 5 concludes the paper.

2 Related Work

The implementation of the e-government concept in the form of administrative procedures, which are automated to a large extent, requires reorganization of all government processes [3]. A step in this direction is process modeling with Business Process Management (BPM) tools [4]. However, to fully understand the processes, it is necessary to adequately represent not only their models, but also the knowledge relating to their execution. On the one hand, execution of administrative procedures requires adequate knowledge, and on the other hand, the execution of administrative procedures is itself a significant source of knowledge. Traditional BPM-based approaches to modeling are not sufficient to describe such knowledge-oriented administrative procedures. Therefore, it became necessary to develop new approaches to enable creating models including all the details reflecting all possible variants of administrative procedures. In these approaches, process models are usually enriched with the business rules and a knowledge base necessary for their operation.

In this respect, a notable example is the approach developed within the FIT project [5]. In this approach, process models include static activities (performed whenever a process is executed) and dynamic activities (performed under certain conditions). A sequence of static activities is planned at design time, while a selection of dynamic activities for execution is carried out at run time based on conditions represented as business rules. In this approach, process models describe typical courses of processes, and rules are used to describe handlers for exceptional situations and unforeseeable events. In the process models, there are knowledge-intensive tasks distinguished, whose execution is dependent on specific circumstances. The processes are modeled in parallel with business rules resulting from legislation and other conditions that may occur at run time. All the circumstances affecting the business rules are represented by concepts defined in an ontology.

Another rule-based approach to modeling and executing of administrative procedures is proposed in [6]. In this work, modeling of administrative procedures is based on composition of elementary processes, which correspond to unit legal aspects. The course of administrative procedures is governed by the provisions of the law that determine what to do depending on various circumstances of specific cases. The course of an administrative procedure for a specific case is dynamically composed from elementary processes based on current legal circumstances occurring during execution of that procedure. The legal circumstances are represented by instances of concepts defined in a business object model (BOM) and recognized by business rules.

The above-mentioned approaches are based on process models, business rules and a knowledge base. The main emphasis is given to modeling of processes and business rules; however, the problem of modeling the relevant knowledge is addressed to a very limited extent. In these approaches, it is mentioned about the various sources of knowledge that may affect the execution of administrative procedures, but there is a lack of a comprehensive architecture of the knowledge base covering all kinds of knowledge that may affect the way the administrative procedures are carried out.

3 Administrative Knowledge Base

The conceptual architecture of the Administrative Knowledge Base for administrative procedure execution is presented in Fig. 1. It consists of three main parts: Business Object Model, Business Rules, and Facts. In unitary states, a single knowledge base can serve the entire state, while in federal states—a single province (land, canton, etc.).

3.1 Business Object Model

Business Object Model defines classes representing objects within the area of focus [7]. In the Administrative Knowledge Base, the BOM includes three categories of classes: legal-based classes, organizational structure classes, and technical infrastructure classes.

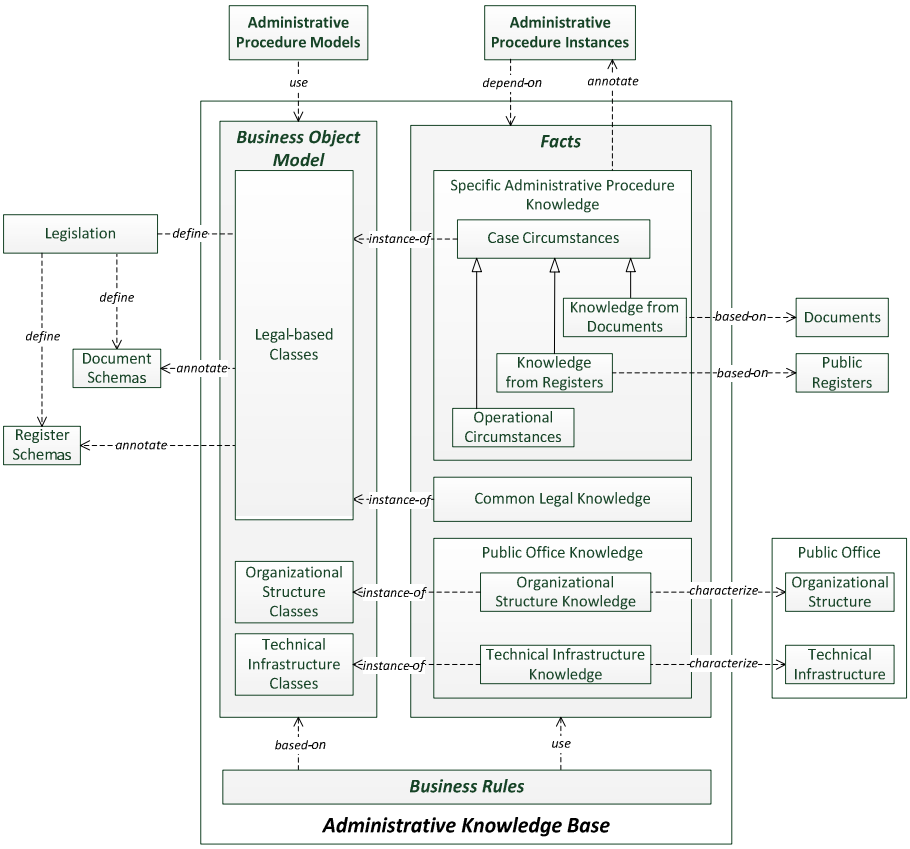


Fig. 1. Conceptual architecture of Administrative Knowledge Base

The legal-based classes result from legislation. They reflect abstract and real objects that appear in the provisions of acts. These are: the classes for mostly abstract objects typically associated with the legal domain, such as legal capacity, i.e. the capacity of individuals to make binding amendments to their obligations, duties and rights; the classes for objects related to the organization and competences of public administration, such as an administrative procedure, administrative proceeding, mayor competence; and the classes for objects related to specific domains regulated by various acts, e.g., a construction site (Building Law Act) or an area being the subject to the environmental protection (Nature Protection Act).

The provisions of legislation also serve as the source material for two other elements: schemas (templates) of electronic documents and schemas of public registers and records. For both, it is necessary to link classes defined in the BOM with elements of the schemas. By doing so, there will be possible to automatically create facts (instances of classes from the BOM) based on data included in documents or retrieved from registers. For documents, it is possible to automatically generate the BOM's classes from XML Schemas using appropriate IT tools; for example, XML Binding Compiler of a framework implementing JAXB specification [8].

3.2 Business Rules

In the Administrative Knowledge Base, business rules have three main areas of application:

- Securing integrity of facts, i.e., checking integrity constraints of facts' attributes and relationships among them;
- Performing tasks in administrative procedures related to analysis of information and inference over the information;
- Analyzing circumstances of administrative procedures to decide on their further course.

The law introduces a great number of restrictions on the relationships between concepts it covers. Many of these restrictions cannot be expressed directly in the BOM due to the lack of a mechanism for imposing constraints on the relationships between classes. Such constraints must therefore be defined some other way; a mechanism perfectly suited to this is business rules. For example, according to the Polish legislation on local government, a state has the following units of administrative division: a civil township, a municipality, a township city, and a municipal town. A municipality is headed by a land mayor, and a civil township is headed by a township governor. A municipal town is headed by a town mayor who is vested with the same executive authority as a land mayor; a township city is headed by a president who is vested with the same executive authority as a land mayor and a township governor. The BOM reflects these issues in a general way, defining the *is-headed-by* relationship between the administrative division unit class and the executive authority class. All constraints between the specific subclasses of an administrative division unit and executive authority classes must be secured with business rules.

The second application area of business rules is data processing and analysis. In public administration, the two main sources of data are: documents and registers. In an ideal situation, documents are delivered as XML files, and registers are maintained as IT systems with on-line access. In such an ideal situation, the content of documents and registers can be used to automatically create facts in the knowledge base. Otherwise, the facts must be created manually by users (clerks), i.e., administrative procedures must include user tasks of entering manually the content of documents or registers into electronic forms.

The third application area of business rules is taking decisions on the further course of administrative procedures, especially selection of subsequent tasks, paths or even whole subprocesses, based on the analysis of the current circumstances of the procedures. For example, if there is an area of Nature 2000 located in the neighborhood of the planned construction investment then it is required by law to request the Regional Directorate of the Environmental Protection for evaluating the impact of this investment on the environment. If the Directorate raises objections it is necessary to notify the applicant about the obligation of introducing corrections to a construction design.

3.3 Facts

Facts are instances of classes defined in the BOM. Facts collected in the Administrative Knowledge Base can be divided into three main areas: common legal knowledge, knowledge specific to a public office, and knowledge specific to particular administrative procedures.

Common Legal Knowledge. This area contains general knowledge resulting from current legislation and independent of a specific public office or administrative procedures where it can be applied. Examples of common legal knowledge are all kinds of classifications and rankings; for example, facts representing units of administrative division of a state: provinces, counties, municipalities, and cities. Facts constituting common legal knowledge are long-lasting; their change results from the change of legislation only.

Public Office Knowledge. This area contains knowledge representing the current organizational and technical circumstances of a specific public office. This knowledge includes:

- Organizational structure of the office. Facts represent organizational units in the office, for example, Department of Urban Planning and Architecture, Motor Vehicle Department;
- Position structure in the office. Facts represent positions and their relationships with the organizational structure, for example, clerk, inspector, director;
- Human resources of the office. Facts represent employees, their personal data and relationships with positions and organizational units, for example Adam Novak, a director of the Department of Urban Planning and Architecture;
- Range of duties. Facts represent relationships between organizational units and administrative procedures which the units were appointed to conduct, for example, Department of Urban Planning and Architecture conducts the procedure of granting a building permit;
- Configuration of IT systems managing public registers, for example, a register of building permits issued is maintained in electronic form, but its content is not accessible in an on-line manner.

Public office knowledge has a semi-permanent character. It changes, for example, due to changes in the organizational structure of the office or due to introduction of new IT systems.

Administrative Procedure Knowledge. This area contains knowledge specific to a particular administrative procedure. It includes facts that are created in the course of this procedure and have specific values resulting from this course, actions taken and other circumstances occurring during the procedure execution. This knowledge includes:

- Knowledge acquired from documents related to the administrative procedure. These are the documents submitted by an applicant, e.g. application, architectural and constructional design; documents created by an office conducting the procedure, e.g. decision, provision; and documents delivered by other public offices and

third parties at the request of the office conducting the procedure, e.g. opinions and statements. If a document is in the XML form then its data can be automatically transformed into facts based on links between classes in the BOM and elements of schema that defines the structure of the document. If a document is in the paper form then it needs to be converted into the XML form, e.g. as a result of manual entering into an electronic form.

- Knowledge acquired from public registers. For example, knowledge retrieved from the Local Development Plan on the existing zoning guidelines for a real property where a new construction is planned. If a specific register is available on-line then a task of acquiring knowledge can be executed automatically, without human (clerk) support. In the case of registers without on-line access or even run in a paper form, it is necessary to generate manual tasks for a user (clerk) to read the relevant piece of data from the register and then enter it into the workflow system carrying out administrative procedures. Such data may be then used to create facts.
- Knowledge related to the operational issues of an administrative procedure execution. This knowledge represents circumstances, especially legal ones that have been identified as a result of performing tasks within the procedure. These tasks relate to the analysis of documents, analysis of the data retrieved from registers, and actions taken by clerks outside the system. For example, if in the Local Development Plan there are no zoning guidelines for a real property where a new construction is planned then the investor is obliged to apply for an outline planning decision—the fact stating the necessity of obtaining such a decision is created. The second type of knowledge related to the operational issues is operational and technical circumstances that exist during execution of an administrative procedure. Examples of such knowledge are: facts reflecting the department and the clerk that led the procedure, facts reflecting technical conditions of IT infrastructure that existed while the procedure was carried out and which resulted in the ability to automatically retrieve data from specific public registers. This knowledge must be separated from the knowledge about the office since the first one represents the specific situation which existed when the procedure was performed, and the latter is a reflection of the current situation in the office, which may change over time.

4 Example of Usage: Granting a Building Permit Procedure

In this section, the application of the Administrative Knowledge Base is presented with an example of an administrative procedure for granting a building permit. This procedure is a universally understandable example for readers from various countries, even if some details may vary.

In the example it is assumed that there is a public office with the following characteristics: (1) legal status: a city hall of a township city; (2) organizational circumstances: organizational unit dedicated to conduct building permit procedures is the Department of Urban Planning and Architecture; (3) IT infrastructure circumstances: a local development plan register is managed by IT system delivering online access through web services. The application for a building permit was filed in an electronic form; there are two attachments to the application: the construction design in an electronic format and the outline planning decision for an area where an investment is planned in the form of a paper document.

4.1 Content of the Administrative Knowledge Base

Fig. 2 presents a fragment of Business Object Model including concepts related to administrative procedure domain.

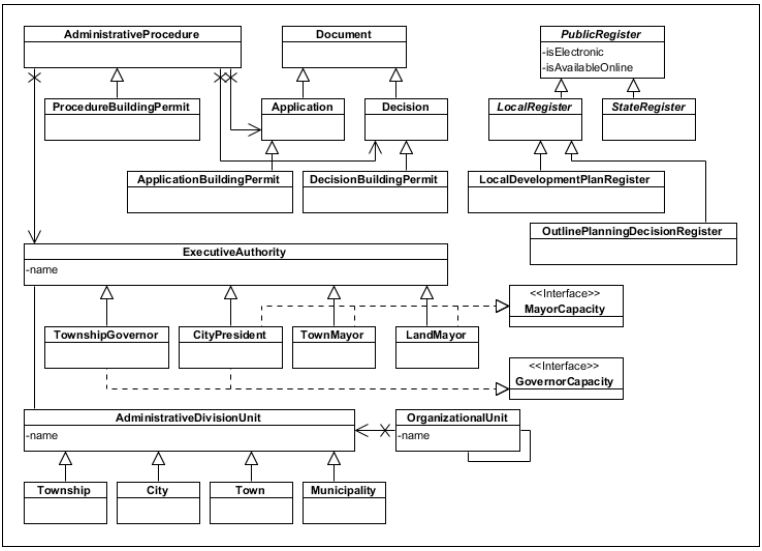


Fig. 2. Business Object Model of administrative procedure concepts

The BOM contains a hierarchy of administrative procedures; one of them is the procedure for granting a building permit, namely the *ProcedureBuildingPermit* class; in the figure, classes of other procedures have been omitted due to the lack of space. Each procedure is connected with at least two documents: an application and a decision. Of course, for every specific procedure the appropriate specific class of the application and decision are necessary. Since it is not possible to satisfy this requirement using constructions available within BOM, it is necessary to apply business rules here. Formula 1 presents the rule implementing this requirement for the building permit procedure. The rules are stored in the appropriate section of the Administrative Knowledge Base.

$$\forall x, y : \text{ProcedureBuildingPermit}(x) \wedge \text{ApplicationBuildingPermit}(y) \wedge \text{application}(x, y) \quad (1)$$

The *AdministrativeProcedure* and *ExecutiveAuthority* classes are associated with each other. Both classes are located at the top of the hierarchies representing specific procedures and specific executive authorities respectively. The rules indicating which specific executive authority is authorized to run a specific procedure are stated in various acts. Same as above, it is necessary to apply business rules to ensure compliance with these rules. The *PublicRegister* class and its child classes represent public registers. Facts based on these classes are used to characterize IT infrastructure available in specific public offices.

Fig. 3 presents a fragment of the Business Object Model including concepts related to an administrative procedure for granting a building permit. The main classes here are: *ProcedureBuidlingPermit* inheriting from *AdministrativeProcedure* and *ApplicationBuildingPermit* inheriting from *Application*. Each of these classes is connected (directly or indirectly through its parental class) with other classes that represent various legal-based circumstances of a building permit procedure; for example, the payment of a stamp duty, the compliance of the application with Building Law Act terminology, the right to use the property for building purposes.

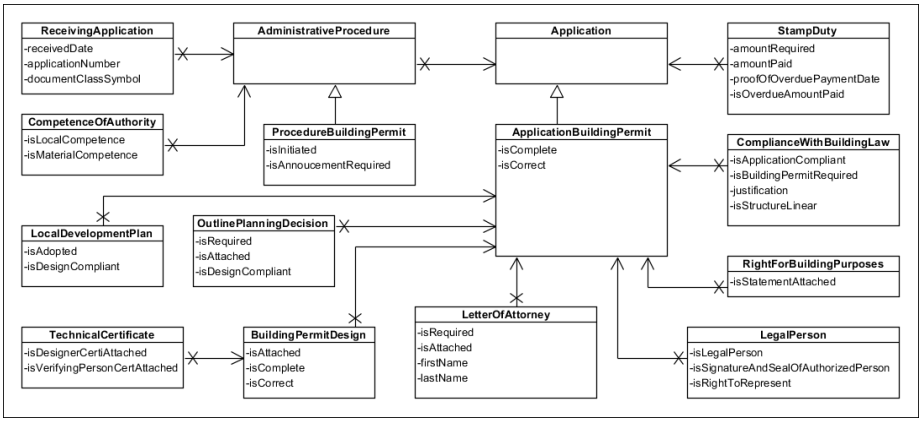


Fig. 3. Fragment of the BOM for an administrative procedure for granting a building permit

Using the presented BOMs, the Administrative Knowledge Base has been populated with the following facts:

- Facts reflecting public office specific knowledge; these are facts of classes as follows: *City*, *CityPresident*, *LocalCompetence*, *OrganizationalUnit*, and *Local DevelopmentPlanRegister*;
- Facts reflecting administrative procedure specific knowledge; these are facts of classes as follows: *ProcedureBuildingPermit*, and *AppicationBuildingPermit*.

4.2 Administrative Procedure Execution

This section presents selected fragments of an administrative procedure for granting a building permit. During the procedure execution, the facts and business rules stored in the knowledge base are used to perform tasks and determine the procedure course.

Verifying Authority's Competence. A model depicting verification of authority's competence is presented in Fig. 4.

Upon receiving an application, it has to be determined whether the public office has local and material competences required to conduct a procedure requested in the application. This determination can be performed automatically based on information included in the application and the knowledge describing legal, technical, and organizational characteristics of the public office. The first task here is *Automatic verifying*

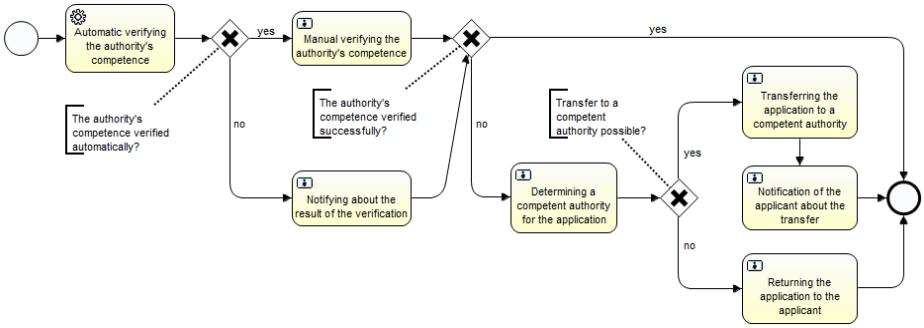


Fig. 4. Verification of authority's competence in the procedure for granting a building permit

the authority's competence. According to the Building Law Act, the material competence for a building permit procedure is held by an office headed by the executive authority having the township governor competences. This is expressed by the rules presented in Formula 2. The rules check if the fact representing the executive authority is of the *GovernorCapacity* class and as the outcome insert the fact of *CompetenceOfAuthority* class with the appropriate value of the *isMaterialCompetence* attribute.

$$\begin{aligned} &\forall x, y : \text{ProcedureBuildingPermit}(x) \wedge \text{GovernorCapacity}(y) \wedge \text{executeBy}(x, y) \Rightarrow \\ &\quad \exists z : \text{CompetenceOfAuthority}(z) \wedge \text{procedure}(x, z) \wedge \text{isMaterialCompetence}(z) \\ &\forall x, y : \text{ProcedureBuildingPermit}(x) \wedge \neg\text{GovernorCapacity}(y) \wedge \text{executeBy}(x, y) \Rightarrow \\ &\quad \exists z : \text{CompetenceOfAuthority}(z) \wedge \text{procedure}(x, z) \wedge \neg\text{isMaterialCompetence}(z) \end{aligned} \tag{2}$$

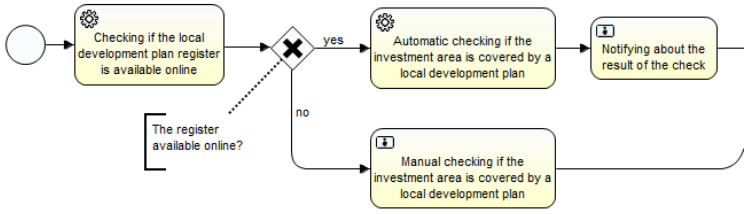
The verification of the local competence in a building permit procedure concerns checking if the investment is located within territorial competences of the authority. Rules implementing this requirement use the fact reflecting the investment location—this fact is created based on data included in the application, and the fact reflecting the territorial scope of the authority's local competences—this fact belongs to the organizational characteristics. The outcome is represented as the value of the *isLocal Competence* attribute of the *CompetenceOfAuthority* class.

If the automatic verification of the authority's competence is possible, then a clerk is notified about the outcome. Otherwise, the authority's competence has to be verified manually based on the application content and the clerk's knowledge.

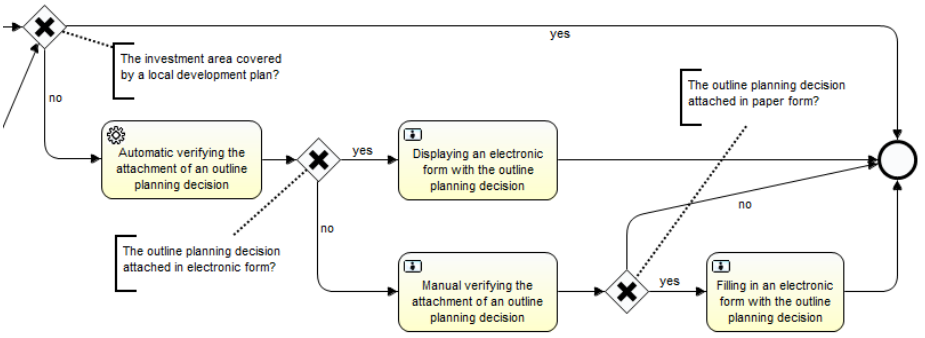
Verifying Outline Planning Decision. A model depicting verification of outline planning decision is presented in Fig. 5. This decision must be attached if the investment area is not covered by a local development plan.

In the first task named *Checking if the local development plan register is available online*, a fact of the *LocalDevelopmentPlanRegister* class is retrieved from the knowledge base in order to check whether the online access to the register is available. If so then the *Automatic checking if the investment area is covered by a local development plan* task in the upper path is picked up. This task is executed in an automatic way using facts in the knowledge base, according to the following logic:

(1) Retrieve from the knowledge base the fact reflecting a location of the investment area; this location is specified in the application and since the application was submitted as an electronic document all its data was inserted into the knowledge base as facts; (2) Check in the local development plan register if the location is covered by a local development plan; (3) Insert into the knowledge base a fact of the *LocalDevelopmentPlan* class representing the information retrieved from the register and a fact of the *OutlinePlanningDecision* class representing the requirement for an outline planning decision.



(a)



(b)

Fig. 5. Verification of outline planning decision in the procedure for granting a building permit

If there is no online access to the register, the procedure is performed along the lower path which includes the task named *Manual checking if the investment area is covered by a local development plan*. The clerk must here manually do the necessary checking and enter the results into the system which in turn leads to the creation of facts in the knowledge base.

Next, if the investment area is covered by the local development plan the upper path is taken and the model flow leads to the end. Otherwise, an outline planning decision is required to be attached and the building design must conform to the provisions of this decision. The administrative procedure is carried out according to the lower path, which starts with the task named *Automatic verifying the attachment of an outline planning decision*. This task is executed automatically according to the following logic: (1) Retrieve a fact reflecting a situation that an outline planning decision is attached as an electronic document; (2) Update the fact of the *OutlinePlanningDecision* class to reflect whether the outline planning decision is attached or not.

If the outline planning decision is attached on paper, its content has to be entered manually into the system. Based on data entered, facts representing information contained in the decision are created in the knowledge base to enable further control of the procedure course based on the provisions of the decision. Also, the fact of the *OutlinePlanningDecision* class has to be updated in order to reflect the attachment of the decision.

5 Conclusions

In this paper, a conceptual architecture of Administrative Knowledge Base for administrative procedure execution has been presented. The knowledge base is intended to collect knowledge necessary to automate execution of routine tasks within administrative procedures performed in public offices. The proposed approach distinguishes three categories of such knowledge: common legal knowledge, public office knowledge, and administrative procedure knowledge. The first one results from current legislation and is independent from public offices and administrative procedures where it can be applied. The second one represents the current organizational and technical circumstances of a specific public office. Finally, the last one is being created during the course of a specific administrative procedure and strictly results from circumstances occurring during the procedure execution. As a proof of a concept, the application of the proposed approach has been illustrated with an example of an administrative procedure for granting a building permit.

References

1. OECD: The E-government Imperative: Main Findings, OECD Observer (March 2003)
2. Strykowski, S., Wojciechowski, R.: Ontology-based Modeling for Automation of Administrative Procedures. In: Grzech, A., Borzemski, L., Świątek, J., Wilimowska, Z. (eds.) Information Systems Architecture and Technology. Service Oriented Networked Systems, pp. 80–97. Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław (2011)
3. Dörfler, A.: Business Process Modelling and Help Systems as Part of KM in e-Government. In: Wimmer, M.A. (ed.) KMGov 2003. LNCS (LNAI), vol. 2645, pp. 297–303. Springer, Heidelberg (2003)
4. Weske, M.: Business Process Management: Concepts, Languages, Architectures, 2nd edn. Springer, Heidelberg (2012)
5. Feldkamp, D., Hinkelmann, K., Thönssen, B.: The Modelling of Knowledge-Intensive Processes Using Semantics. In: Vitvar, T., Peristeras, V., Tarabanis, K. (eds.) Semantic Technologies for E-Government, pp. 75–98. Springer, Heidelberg (2010)
6. Strykowski, S., Wojciechowski, R.: Composable Modeling and Execution of Administrative Procedures. In: Kő, A., Leitner, C., Leitold, H., Prosser, A. (eds.) EDEM 2012 and EGOVIS 2012. LNCS, vol. 7452, pp. 52–66. Springer, Heidelberg (2012)
7. Del Fabro, M.D., Albert, P., Bézivin, J., Jouault, F.: Achieving Rule Interoperability Using Chains of Model Transformations. In: Paige, R.F. (ed.) ICMT 2009. LNCS, vol. 5563, pp. 249–259. Springer, Heidelberg (2009)
8. Java Architecture for XML Binding Compiler (xjc), `http://jaxb.java.net/2.2.4/docs/xjc.html`

Author Index