Witold Abramowicz (Ed.)

Business Information Systems

LNBIP 21

12th International Conference, BIS 2009
Poznań, Poland, April 2009
Proceedings

Springer

Lecture Notes in Business Information Processing 21

Witold Abramowicz (Ed.)

Business Information Systems

12th International Conference, BIS 2009
Poznań, Poland, April 27-29, 2009
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

Library of Congress Control Number: Applied for

ACM Computing Classification (1998): J.1, H.3.5, H.4, D.2.12

ISSN 1865-1348
ISBN-10 3-642-01189-6 Springer Berlin Heidelberg New York
ISBN-13 978-3-642-01189-4 Springer Berlin Heidelberg New York

springer.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 12659579 06/3180 5 4 3 2 1 0

Preface

BIS 2009 held in Poznań was the 12th in a series of international conferences on Business Information Systems. The BIS conference series has been recognized by professionals from its very beginning as a forum for the exchange and dissemination of topical research in the development, implementation, application and improvement of computer systems for business processes.

The theme of the conference was "Information Systems for Agile Organizations." The material collected in this volume covers recent tendencies in making organizations more responsive to external conditions. On the one hand, ontologies are adopted as a tool for better data and information management. On the other hand, flexible information systems based on process orientation are being developed. A set of 22 papers illustrating these trends were selected for presentation during the main event. The articles are grouped into the following conference topics: Ontologies in Organizations, Ontologies and Security, Process Modelling, Process Analysis and Mining, Service-Oriented Architecture, Web Search, Social Issues, and ERP.

The Program Committee consisted of over 90 members, who carefully evaluated all the submitted papers. The careful review process assured that each paper underwent thorough evaluation and received many valuable comments. Only the best-quality papers were selected, resulting in an acceptance rate of less than 30%.

The regular program was complemented by outstanding keynote speakers. We were honored to host leading experts in business process management and information management: Wil van der Aalst (Eindhoven University of Technology, The Netherlands) and Asunción Gómez-Pérez (Universidad Politécnica de Madrid, Spain).

BIS 2009 was kindly supported by Service Web 3.0 (FP7-216937) and Semantic Technology Institute International (Austria).

April 2009 Witold Abramowicz

Conference Organization

BIS 2009 was Organized by

Poznań University of Economics, Department of Information Systems.

Program Chair

Witold Abramowicz

Program Committee

Harith Alani	University of Southampton, UK
Dimitris Apostolou	University of Piraeus, Greece
Djamal Benslimane	Universite Claude Bernard Lyon, France
Diego Calvanese	Free University of Bozen-Bolzano, Italy
Gerardo Canfora	University of Sannio, Italy
Longbing Cao	University of Technology, Australia
Jorge Cardoso	SAP Research, CEC Dresden, Germany
Wojciech Cellary	Poznań University of Economics, Poland
Houn-Gee Chen	National Taiwan University, Taiwan
Huajun Chen	Zhejiang University, China
Dickson K.W. Chiu	Dickson Computer Systems, Hong Kong
Oscar Corcho	Universidad Politechnica de Madrid, Spain
Flavio Corradini	University of Camerino, Italy
Sergio de Cesare	Brunel University, UK
Tharam S. Dillon	Curtin University of Technology, Perth
Ying Ding	Indiana University, USA
Marlon Dumas	University of Tartu, Estonia
Schahram Dustdar	Vienna University of Technology, Wien
Martin Dzbor	The Open University, UK
Dieter Fensel	University of Innsbruck, Austria
Bogdan Franczyk	University of Leipzig, Germany
Johann-Christoph Freytag	Humboldt University, Germany
Marko Grobelnik	The Jozef Stefan Institute, Slovenia
Hele-Mai Haav	Tallinn University of Technology, Estonia
Mohand-Said Hacid	Universite Claude Bernard Lyon, France
Axel Hahn	University of Oldenburg, Germany
Martin Hepp	Bundeswehr University Munich, Germany
Knut Hinkelmann	University of Applied Sciences Northwestern Switzerland, Switzerland

Jack Hsu	National Sun Yat-Sen University, Taiwan
Marta Indulska	The University of Queensland, Australia
Pontus Johnson	Royal Institute of Technology, Sweden
Paweł J. Kalczyński	CSU Fullerton, USA
Uzay Kaymak	Erasmus University Rotterdam, The Netherlands
Jerzy Kisielnicki	University of Warsaw, Poland
Gary Klein	University of Colorado at Colorado Springs, USA
Ralf Klischewski	German University in Kairo, Egypt
Jacek Koronacki	Polish Academy of Sciences, Poland
Sanjay Kumar	XLRI School of Management, India
Dennis Kundisch	Albert-Ludwig University of Freiburg, Germany
Dominik Kuropka	University of Potsdam, Germany
Peter C. Lockemann	University of Karlsruhe, Germany
Peter Loos	University of Saarbrucken, Germany
Ruopeng Lu	SAP Research, Australia
Zakaria Maamar	Zayed University, United Arab Emirates
Leszek Maciaszek	Macquarie University, Australia
Qusay H. Mahmoud	University of Guelph, Canada
Yannis Manolopoulos	Aristotle University, Greece
Jorge Marx Gómez	University of Oldenburg, Germany
Yutaka Matsuo	University of Tokyo, Japan
Florian Matthes	Technical University of Munich, Germany
Heinrich C. Mayr	University of Klagenfurt, Austria
Marie-Francine Moens	Katholieke Universiteit Leuven, Belgium
Günter Müller	University of Freiburg, Germany
John Mylopoulos	University of Toronto, Canada
Adam Nowicki	The Wroclaw University of Economics, Poland
Andreas Oberweis	University of Karlsruhe, Germany
Mitsunori Ogihara	University of Miami, USA
Marcin Paprzycki	Polish Academy of Sciences, Poland
Eric Paquet	National Research Council, Canada
Vassilios Peristeras	National University of Ireland, Ireland
Jakub Piskorski	EU, JRC, Web and Language Technology, Italy
Jaroslav Pokorný	Charles University, Czech Republic
Elke Pulvermueller	University Osnabrueck, Germany
Hajo A. Reijers	Eindhoven University of Technology, The Netherlands
Ulrich Reimer	University of Konstanz, Germany
Riccardo Rosati	Universita di Roma, Italy
Gustavo Rossi	National University of La Plata, Argentina
Shazia Sadiq	The University of Queensland, Australia
Jurgen Sauer	University of Oldenburg, Germany

Alexander Schill	Dresden University of Technology, Germany
Elmar J. Sinz	University of Bamberg, Germany
Jaideep Srivastava	University of Minnesota, USA
Kilian Stoffel	University of Neuchatel, Switzerland
Heiner Stuckenschmidt	Universitat Mannheim, Germany
Vojtech Svatek	VSE Praha, Czech Republic
David Taniar	Monash University, Australia
Sergio Tessaris	Free University of Bozen - Bolzano, Italy
Barbara Thönssen	University of Applied Sciences Northwestern Switzerland, Switzerland
Robert Tolksdorf	Free University Berlin, Germany
Olegas Vasilecas	Vilnius Gediminas Technical University, Lithuania
Herna Viktor	University of Ottawa, Canada
Hans Weigand	Tilburg University, The Netherlands
Mathias Weske	Hasso Plattner Institute for It-Systems Engineering, Germany
Krzysztof Węcel	Poznan University of Economics, Poland
Roel Wieringa	University of Twente, The Netherlands
Stanislaw Wrycza	University of Gdansk, Poland
Hui Xiong	Rutgers, The State University of New Jersey, USA
Seiji Yamada	National Institute of Informatics, Japan
Yun Yang	Swinburne University of Technology, Australia
Yiyu Yao	University of Regina, Canada
Slawomir Zadrozny	Polish Academy of Sciences, Poland
John Zeleznikow	Victoria University, USA
Shuigeng Zhou	Fudan University, China
Jozef Zurada	University of Louisville, USA

Organizing Committee

Elżbieta Bukowska	Poznań University of Economics, Poland
Dominik Flejter	Poznań University of Economics, Poland
Konstanty Haniewicz	Poznań University of Economics, Poland
Lidia Leszczyńska	Poznań University of Economics, Poland
Jolanta Skorwider	Poznań University of Economics, Poland
Monika Starzecka	Poznań University of Economics, Poland
Piotr Stolarski	Poznań University of Economics, Poland
Krzysztof Węcel	Poznań University of Economics, Poland
Adam Walczak	Poznań University of Economics, Poland

External Reviewers

Algirdas Laukaitis
Andre van Cleeff
Barbara Re
Bo Liu
Bruno Pouliquen
Carlos Buil
Corrado Aaron Visaggio
Daniela Feldkamp
Diletta Romana Cacciagrano
Dong Yuan
Elena Montiel-Ponsoda
Emilian Pascalau
Eugenio Zimeo
Francesco de Angelis
Harald Meyer
Holger Wache
José Manuel Gómez-Pérez
Lemonia Giantsiou
Leonardo Vito
Manuel Salvadores
Martin Szomszor
Maya Daneva
Nikos Loutas
Oliviero Riganelli
Pascal van Eck
Per Närman
Raúl García-Castro
Sergey Smirnov
Shanshan Wu
Shouichi Matsui
Stephan große Austing
Teodor Sommestad
Tom Broens
Xiao Liu
Xiaohui Zhao
Yanshan Xiao
Zhigang Zheng
Ziye Zuo

Table of Contents

Ontologies in Organizations

Ontologies and Security

Web Search

Social Issues

Process Modelling

Process Analysis and Mining

Service-Oriented Architecture

ERP

Organisational Ontology Framework for Semantic Business Process Management

Agata Filipowska[1], Martin Hepp[2], Monika Kaczmarek[1], and Ivan Markovic[3]

[1] Department of Information Systems, Poznan University of Economics
61-875 Poznan, Poland
{a.filipowska,m.kaczmarek}@kie.ue.poznan.pl
[2] Chair of General Management and E-Business, Bundeswehr University Munich
88579 Neubiberg, Germany
mhepp@computer.org
[3] SAP Research, 76131 Karlsruhe, Germany
ivan.markovic@sap.com

Abstract. The field of Semantic Business Process Management (SBPM) has refuelled interest in using ontologies for the representation of the static and dynamic aspects of an enterprise and value chains. Putting the SBPM vision into practice, however, requires a consistent and operational network of ontologies reflecting the various spheres of enterprise structures and operations. Consistent means that the ontologies are based on compatible paradigms, have a compatible degree of detail, and include at least partial sets of alignment relations which allow data interoperability. Operational means that the ontology specifications are available in a single, current ontology formalism for which scalable repositories, reasoning support, APIs, and tools are available. In this paper, we describe a set of ontologies for SBPM that follows the mentioned requirements, and compare our work with the related efforts.

Keywords: Ontology, Business Process Management, enterprise description, organisation framework.

1 Introduction

Although BPM used together with SOA is believed to provide a complex approach to manage business processes in an enterprise, currently it offers only little support for automation of the BPM lifecycle. It is especially visible when it comes to the difficulties in smooth and automatic transition from one phase to another.

Various attempts were undertaken to provide a holistic view of the process space and achieve a higher level of automation in the BPM lifecycle. One of the most advanced initiatives in this area is the concept of Semantic Business Process Management (SBPM) developed in the SUPER project [1] taking advantage of the Semantic Web technologies (ontologies and reasoning mechanisms).

In order to fulfil its aims, SBPM needs a semantic representation of various artefacts used within all of the phases of business process management. Therefore, apart from the semantic description of the process flow (control structure), the process

W. Abramowicz (Ed.): BIS 2009, LNBIP 21, pp. 1–12, 2009.

content description is also required. The process content relates to the enterprise and its environment. Thus, putting the SBPM vision into practice requires a consistent and operational network of ontologies reflecting the various spheres of enterprise structures and operations. Consistent means that the ontologies are based on compatible paradigms, have a compatible degree of detail, and include at least partial sets of alignment relations which allow data interoperability. Operational means that the ontology specifications are available in a single, current ontology formalism for which scalable repositories, reasoning support, APIs, and tools are available.

The goal of this paper is to present the notion of the consistent and operational set of organisational ontologies aiming at providing vocabulary and constraints for describing the environment in which processes are carried out from the organisations perspective. In order to fulfil its aims, the article is structured as follows. First, the related work in the area of process artefacts representation is shortly discussed. In the following section the description of a consistent organisational ontology stack is given along with an application of the organisational ontologies within a domain-specific use case scenario. In Section 4 we conclude our work with a comparison between our work and related approaches and provide arguments in favour of our contribution.

2 Related Work

The process content depends on organisation and its characteristics. Many researchers focused on development of models or ontologies cataloguing an appropriate scope of information on companies that should be considered when describing organizations or their processes e.g. [2][3]. For instance [2] highlighted that a process model should provide information on what is going to be done (i.e. functionality), who is going to do it (i.e. actors), when and where it will be done (i.e. location and time), how and why it will be done (i.e. means and motivation), as well as who depends on the results that are to be achieved (i.e. dependencies with other processes). Further on, the authors distinguished four different perspectives on the process, namely: functional, behavioural, organisational and informational. These perspectives underlie separate yet interrelated representations for analyzing and presenting processes.

The mentioned initiatives focused not only on the scope of information to be considered but also on its representation. Within the last few years there have been numerous initiatives attempting to capture the process-related and also organisation-related information in a machine-friendly manner. Most of them focused on possible application of semantics, as ontologies are perceived as a good way to capture the domain and its relations [4, 5, 10, 15]. These initiatives differ, when it comes to the scope of the process description they intend to cover, the level of details of the ontology created, as well as the formalism used.

One of the earliest initiatives was the TOVE project [6] that aimed at development of a set of integrated ontologies for modelling all kinds of enterprises (i.e. commercial and public ones) [4]. TOVE Common Sense Model of Enterprise included three levels: reference model with typical business functions (finance, sales, distribution, and administration), generic model (with such concepts as time, causality, space, etc.), and concept model (e.g. role, property). However, the granularity of developed ontologies may be perceived inconsistent what hampers their potential application.

The REA enterprise ontology [7] is based on elements of the REA (Resource-Event-Actor) model [8]. The REA concepts and definitions are applied to the collaborative space between enterprises where the market exchange among trading partners occurs. Although REA is considered one of the most promising business domain ontologies, it is criticized for the lack of clarity and inconsistence e.g. [9].

The main aim of the e3-value project [10] was to propose the methodology to help in eliciting, analyzing, and evaluating e-commerce ideas. Therefore, the e3-value ontology was introduced as a tool to help explaining the concepts used to represent e-commerce ideas. The ontology provides concepts for describing economic exchange among partners. Other e3-value extensions like e3forces, e3competences should be of particular attention as they intend to model more advanced organisational aspects.

Enterprise Ontology (EO) [5] is a collection of terms and definitions relevant to business enterprises. It was developed as part of the Enterprise Project, with the aim to provide a framework for enterprise modeling. EO is divided into five parts: i) terms related to processes and planning, ii) terms related to organisational structure, iii) terms related to high-level planning, iv) terms relating to marketing and sales and v) terms used to define the terms of the ontology together with terms related to time. It was first completed in natural language format and then ported to Ontolingua [5].

Although significant effort was already committed to the creation of business and enterprise ontologies and the mentioned initiatives may provide an inspiration and foundation for developing organisational ontologies, to the best of our knowledge, there is no commonly accepted model that could be reused in various domains.

3 Ontology Framework

The semantic process representation required by SBPM may be divided into three main groups, namely; process, organisational and domain-specific ontologies. Process ontologies [14] are created in order to describe the structure of a process, whereas organizational ontologies provide a description of artefacts, actors etc. that are utilised or involved in the process. The domain ontologies provide the additional information specific to an organisation from a given domain.

The organisational ontologies, this paper focuses on, aim at providing vocabulary and constraints for describing the environment in which processes are carried out from the organisations' perspective. Following [15], the organisational ontologies provide a basic vocabulary and structure for describing organisations, business goals and resources, define common types of divisions, roles and tasks, and define common types of business resources. Thus, the organisational ontologies layer provides a high level view on the organisation and process-related space and may be logically divided into a few subontologies, each of them describing different part of this space:

1. **Organisational Structure Ontology (OSO)** focusing on organisational structure (hierarchy) of a company. It is designed as an upper level ontology and provides main structure and relations aiming at achieving domain independency.
2. **Organisational Units Ontology (OUO)** providing specification of typical units that may be found in a company. Along with Business Functions, Business Roles and Business Resources Ontologies, it provides extensions to OSO.

3. **Business Roles Ontology (BROnt)** representing roles in the organisation e.g. Designer, Process Modeller, IT Expert, CEO.
4. **Business Functions Ontology (BFO)** provides hierarchy of different functions that may be carried out within the company. It is supposed to enable vendor and domain independent classification of company processes and process fragments providing abstraction over single tasks constituting processes.
5. **Business Resources Ontology (BRO)** describing resources spent when carrying out certain processes or that may be results of certain task in a process.
6. **Business Goals Ontology (BGO)** modelling a hierarchy of business goals and provides a set of relations between them to enable goal-based reasoning.

In next sections of this paper each of the ontologies being a part of the organisational layer is presented. Due to the different character and status of the ontologies in question there is a difference in the level of details given while presenting them. They all have been modelled using WSML [25] in its Flight variant as the representation language.

3.1 Organisational Structure Ontology and Organizational Units Ontology

An organisational structure is defined as a hierarchy of an organisation showing how its elements work together in order to achieve organisation's goals. Following [17] the organisation structure encompasses: departments, employees, their responsibilities, resources etc. as well as relations among them.

The Organisational Structure Ontology (OSO) focuses on organisation hierarchy. The OSO structure benefited from already described models [4], [5], [12], [13] and [18]. The main distinguished concepts are: organisation, legal and non-legal entity, organisational unit, business function, person, skills, role and organisational position as well as resource (for exact definitions please refer to [16]).

It was designed as an upper level ontology providing the main domain-independent structure and relations. The structure of OSO makes it easy to be imported by other ontologies constituting the organisation ontology. And so Business Roles Ontology refers to the Role concept, Business Functions Ontology refers to the Business Function concept, Resource Ontology refers to the Resource, and finally Organisational Units Ontology refers to the Organisational Unit concept. OSO enables full description of an organisational structure and in addition links the mentioned organisational ontologies.

In turn, in case of the Organisational Units Ontology, an organisational unit is defined as any recognized association of people in the context of an enterprise. Following [17], it may be a corporation, a division, a department, a group or a team as well as a committee, task force, or a class [13].

As we decided to follow the OMG definition of Organisational Units [13], all units are divided into temporary and permanent ones. Temporary units are entities, that are created in order to carry out a task, project etc. They exist in an organisational structure only as long as a task is carried out. Permanent Units included in the structure are common for many organisations and were selected as a result of analysis of different organisational structures of existing companies as well as SAP Solution Maps [20]. A variety of existing departments as well as naming standards, forced us to use some simplification. In consequence, the developed ontology includes only the most common organisational departments in a production enterprise. Therefore, in order to describe some highly specific organisational units of e.g. security company, additional concepts need to be defined.

3.2 Business Roles Ontology

A business role is defined a set of expected behaviours, prerogatives and obligations featured by an actor [12]. The Business Roles Ontology provides a common meaning of concepts related to roles featured by organisational members. Each actor may play more than one role and these roles may change depending on the context.

The BRO was designed with an aim to provide a domain-independent, yet comprehensive, set of concepts, that would allow for description of roles in the organisation microenvironment i.e. both the inside structure of an organisation and its close surroundings. The proposed ontology allows to model both internal and external interactions and consists of 37 concepts. The fundamental concept for the ontology is a BusinessRole. The next level of the model splits up into three concepts. Two of them being specialized BusinessRole: InternalRole and ExternalRole. The third one – InternalRoleType – provides the possibility to extend the description of any InternalRole. ExternalRole concept represents any BusinessRole that characterizes an agent from the outside of the organisation (e.g. BusinessPartnerRole further divided into IntermediaryRole and ProcurerRole). The assumption about ExternalRoles is that they are not disjoint. In contrast to ExternalRole, InternalRole is defined as BusinessRole that characterizes an agent from within the organisation (e.g. CustomerServiceRole). Any two InternalRoles are not mutually exclusive.

The current version of the Business Roles Ontology is supposed to answer the following exemplary competency questions: what role(s) does an actor or a group of actors play in the organisation? How many roles are featured by an actor or a group? Are there other actors with the same or similar role? What are the main and additional roles? Are the roles internal or external? Are there any specific (expanding) features characterizing a given role?

3.3 Business Functions Ontology

A business function is understood as a functional area of an enterprise e.g. Sales Management, Risk Management, Customer Management. The Business Functions Ontology (BFO) aims at standardizing meaning of business concepts and thus provides a common vocabulary for business functions within enterprises [18]. It was designed as a reference ontology, therefore, provides structure of business functions that are common for every company, regardless of the domain it operates in. It acts like an upper ontology for enterprise specific functional ontologies. It is supposed to be a starting point for further development of the business functions and should be extended with domain-specific functions.

Concepts included in the BFO as well as its structure are a result of analysis of existing ERP systems and abstracting over the SAP Business Maps [20]. While designing the BFO, the main goal was to provide as much information as possible and at the same time keep the ontology industry-independent.

BFO consists of two main structures [18], namely: Function and ActivityOrStep. The structure *Function* is the high level view on the functional side of the enterprise. It is specialised by such concepts as e.g.: *Customer Management, Marketing Management* and *Sales Management*. The Function structure contains only 40 concepts, 14 of them are top level, and the rest is grouped as their sub-concepts. The top level concepts name coherent and in some degree autonomous areas of functionalities. In addition, some of the top concepts such as e.g. FinanceManagement, are divided into sub-functionalities.

The *ActivityOrStep* structure supplements the Function structure and presents a more detailed view on the performed functions. An activity is understood as a unit of work identified within the business process e.g. UpdateSalesOrder. ActivityOrStep structure was designed at far more detailed level of abstraction and contains 920 concepts grouped as sub-concepts of 33 top-level concepts. Function and ActivityOrStep structures are connected via a transitive isSubPhaseOf attribute of each of top level concepts from ActivityOrStep sub-structure which determines which concept from Function structure is complemented by particular group of activities and steps. Appropriate axioms ensure the correctness of the specified relations.

The current version of the Business Functions Ontology is to answer the following competency questions: what are the main functional areas of the enterprise? how can the areas be further divided? what activities does the X-business function include? what kind of business function is an X- business sub-function? Within which functions the given activity or step is performed?

3.4 Business Goals Ontology

In our work we define business goal as *the state of the enterprise that an action, or a given set of actions, is intended to achieve* [24]. The Business Goals Ontology provides a standard set of properties and relations used in modelling a hierarchy of organisational business goals and enables formal verification of goal specifications.

Depending on the time horizon, business goals may be classified as operational or strategic. A strategic goal tends to be longer term and defined qualitatively rather than quantitatively. An operational goal is a short-term contribution to a strategic goal and provides the basis for measuring the progress toward meeting strategic goals.

Goals may be quantitative or qualitative. Quantitative goals are specified in a way allowing devising (measuring) the state of the goal by comparing values stated in the goal that express desired state of the world and the actual state. Qualitative goals are using textual descriptions of the desired state, and verification of the state of the goal requires human judgement. Quantitative goals need to have measures and threshold values defined, qualitative goals do not have to have such properties provided.

Business goal can have a more detailed *description*, it has an assigned *measure* for controlling the progress, *deadline* for completion and *priority*. In addition, as we can only control what we can measure [19], we assign a *Measure* to each quantitative goal. Each *Measure* has a defined *Unit*. *Measure* has a *Current Value* in the observed moment and a *Desired Value* which should be reached by a goal.

An important notion to keep in mind when talking about goals is time – goals have to be achieved in a certain time period (deadline). Goals can have priorities, which allow us to order the goal set according to their importance. Prioritization focuses attention on key areas, while allowing the modeler to describe goals which are currently perceived as less important or out of scope. Analysts can choose to show only those goals that are in scope. The goal hierarchy approach thus makes clear the effects of scoping decisions, and allows trade-offs to be evaluated.

Business goal models usually contain a hierarchy of an organisation's business goals according to which the processes in the organisation are designed. The higher level goals are refined into more specific goals through AND or OR-decomposition, where the goals are decomposed into alternative subgoals (OR) or combinations of subgoals that must be satisfied (AND). The relation *subgoal_of* together with the constructs for AND/OR goal decomposition is used when creating the goal hierarchy. The relation *subgoal_of* is transitive, non-reflexive and anti-symmetric. An *atomic goal* is a goal that can not be further divided into subgoals.

Additional goal influencing relation types can exist. A support relationship between goals g_1 and g_2 suggests that achievement of goal g_2 assists achievement of goal g_1; however achievement of goal g_2 is not a necessary condition for achievement of goal g_1, or else goal g_2 would be a successor goal of goal g_1. On the other hand, a hinders relationship between goals g_1 and g_2 suggests that achievement of goal g_1 negatively influences achievement of goal g_2. Relations supports and hinders are transitive, non-reflexive and anti-symmetric. Goals can also conflict with each other - if a goal g_1 hinders a goal of higher priority g_2, goals g_1 and g_2 are in conflict.

Introducing formalized business goal models enables new possibilities in process analysis [24]. For example the following queries can be executed using the business goals ontology: Show me all processes that support a specific goal. Show me the goal that is supported by the specific process. Show me all goals that are not completely specified. Filter goals on the basis of a given deadline and/or priority. Show me all goals that have no business process linked. Show me all processes that do not support any goal (gap analysis). Show me all processes that hinder the achievement of a specific goal. Show me all conflicting/redundant goals.

3.5 Business Resource Ontology

When formalizing knowledge about the business process space of an organisation, we also need a mean for describing the tangible and abstract resources that are relevant in the respective operations. For example, we need to know which tools are required for a particular production step. Therefore, we propose a *Business Resources Ontology*, which defines the core abstractions for resources, and associated notions of access, ownership, and consumption. This ontology can be imported and refined in more specific domain ontologies that define common types of business resources for specific domains, e.g. such things, e.g. *WeldingStation or GalvanizationVessel.*

We reuse the definition by Uschold et al, saying that a resource is „something that can be used or consumed in an activity" [5, p. 41], as the most mature and conceptually clean specification of resources. Since being a resource is a role of an object and not a subtype of an object, the SUPER Resources Ontology does not contain a concept *resource*, but a relationship *requiresAsResource.*

This may be surprising but is a consequence of the fact that being a resource actually *relates* an object to an activity. In WSML syntax, this relation can be specified as follows: **relation** requiresAsResource (**ofType** Activity, **ofType** Object, **ofType** TypeOfPropertyRight, **ofType** _boolean); where:

- requiresAsResources means that the execution of the given Activity requires access in the form of the specified TypeOfPropertyRight to the specified Object;
- Activity is a subconcept of the AcitvityOrStep concept from the Business Function Ontology (i.e. bfo#ActivityOrStep);
- Object is the explicitly identified or anonymous object needed. The characteristics of the object are specified by the respective class of object;
- TypeOfPropertyRight is the type of property right that is needed to specify what type of access is needed on that resource. For example, we may want to specify whether the object is modified, transformed, or even destroyed when serving as a resource, or whether it is just temporarily used. Economic theory and law uses the notion of property rights for distinguishing such types of access to a good. We use the four core distinctions from Property Rights theory. When we want to check whether a particular resource is available, we need a mean to specify what types of property rights are assigned to the entity who wants to use the respective object. This can be done best using a ternary relation hasPropertyRightOnObject (ofType Entity, ofType Object, ofType TypeOfPropertyRight);
- the boolean attribute indicates whether using the resource for the given activity consumes the resource or does not consume it. This is an important distinction, since some resources are lasting (e.g. documents, machinery) while others are consumed in the usage (e.g. coal, fuel).

Since we have no concepts for resources, the characteristics of an object serving as a resource must be reflected by the definition of the object and not its resource role. The following competency questions describe the scope of this subontology: which resources/types of resources exist in an enterprise? Which resources are required for a particular task? Which individual or legal entity has property rights on a particular resource? Does the regular usage of a given resource consume this resource?

3.6 Use Case Study

This section provides an example of application of organisational ontologies within the SUPER project based on a Digital Asset Management use case scenario in the telecommunications domain. The aim of this scenario is the provision of digital content to end users. The service provider, based on a customer query on the digital content specifying also the type of digital rights license the user is interested in, searches for the relevant digital content by several content providers. The search results are then presented to the user, who selects the content he wants to preview or purchase.

Telco ontologies used in this scenario were created to extend the organisational ontologies and provide domain specific concepts which are supplementary to the basic domain-independent concepts, while following the design principles provided by the organisational ontology layer. Further details on the development of telco ontologies within the SUPER project are provided in [22].

In the following, we provide some example queries which demonstrate the usage of our ontology framework within the use case scenario. The queries are expressed using WSML-Flight logical expressions, where boldfaced keywords refer to keywords in the WSML language.

Query 1. Show me all business goals with high priority:

```
?goal[bgo#hasPriority hasValue high] memberOf
bgo#BusinessGoal
```

Query 2. Show me all business goals in conflict with the goal ImproveServices:

```
bgo#hinders (?g, ImproveServices) and
bgo#priority_lt(?g, ImproveServices)
```

The following example demonstrates how we use our integrated view on the process space to query different process perspectives:

Query 3. Show me all processes which have Business Function related to 'Fulfilment' and use system 'CRM' as the Business Resource:

```
?process [bpo#hasBusinessFunction hasValue ?bf and
bpo#hasBusinessResource hasValue ?bres ] memberOf
bpo#Process and ?bf memberOf telbfo#Fulfilment and
?bres memberOf telbro#CRM
```

Notice that *bpo* refers to the previously mentioned process ontology. Symbols *telbfo* and *telbro* denote the telco domain specific extensions of the Business Functions Ontology and Business Resource Ontology, respectively.

Query 4. Within which functions the activity ValidateOrder is performed?

```
?function[telbfo#isSubPhaseOf hasValue ValidateOrder]
memberOf telbfo#BusinessFunction
```

Note that in answering this query we utilize the transitivity axiom of the *isSubPhaseOf* relation.

Query 5. Show me all processes that support the business goal IncreaseRevenue:

Here we first need to find which business goals support the goal *IncreaseRevenue*:

```
telbgo#supports(?g, IncreaseRevenue)
```

where symbol *telbgo* represents the telco domain extension of the Business Goals Ontology. In order to answer our query we need to find processes which are annotated with the supporting business goal *g*:

```
?process [bpo#hasBusinessGoal hasValue g] memberOf
bpo#Process
```

Querying of organisational ontologies has been successfully utilized in various scenarios within the use case, such as: decision making support, reuse of business artefacts in process modelling and process redesign, see [23, 24, 26] for more details.

4 Discussion and Conclusions

In this section we compare the organisational ontologies presented in this article with the previous attempts in this area taking into account the defined criteria, namely: comprehensiveness, consistency and operational aspects (see Table 1).

Table 1. Comparison of the approaches

	Comprehensiveness		Consistency	Operational aspects	
Approach	**Domain covered**	**Easy Extensible**	**Comparable level of details**	**Ontology language used**	**Scalable reasoner etc. availability**
TOVE	Generic enterprise model (Conceptualizations of: agents, roles, positions, goals, communication, authority, commitment thus it models an organisational structure, activities and management.)	-	-	FOL representation	+/-
REA	Creation of transfer of economic value. The core concepts are: Resource, Event, and Actor.	-	-	None commonly accepted	-
e3-value	Identifying exchange of value objects between the business actors. The basic concepts in e3-value are actors, value objects, value ports, value interfaces, value activities and value exchanges.	+/-	+/-	MODEL description logic language	+/-
Enterprise Ontology	Organisational structure, activities and management. Four subject areas are distinguished: activities and planning, organisation, strategy, marketing	+/-	+	Informal (text) and semi-formal (Ontolingua)	-
SUPER	Organisation context with main focus on functions, goals, organisation structure, roles, resources.	+	+	WSML	+

The approaches have usually different focus and the domain covered is not always comparable. While the TOVE and EO projects focus on providing generic enterprise model, REA and e3 value focus more on creation and transfer of economic value, our approach aims at describing organisation context relevant for the needs of business process modelling. The already developed ontologies cover often a substantial area of knowledge, however their usefulness is sometimes limited as the modelled parts of information space are defined on the different level of details and are sometimes inconsistent. In addition, only few initiatives define alignments between various ontologies constituting the set. For instance in TOVE project, three distinguished layers were designed at an inconsistent level of details (inconsistent granularity) and in case of REA the terminology for constructs is criticized for the lack of clarity [9]. The set of ontologies proposed for the needs of SUPER has been aligned and constitutes a consistent and coherent stack, as we have shown earlier in this paper.

In addition, the proposed approaches are rather domain-independent and general in essence, thus in order to improve their usability they should be considerably extended with a domain-specific vocabulary what sometimes is impossible as only few of them were designed bearing in mind further extension. In opposition to these approaches, our ontology stack was designed with the aim to achieve domain-independence as well as allow for easy extensions and usage of domain and industry specific ontologies within the framework as presented in the previous section.

Moreover, the lack of the formal version (expressed using one of the ontology languages) hampers practical application of some of the already defined ontologies (e.g. EO). Even if the ontology are presented in a more formal way (e.g. TOVE), very often the results are not serialized in any contemporarily recognized ontology language standard for which an efficient reasoner exists. In our case, usage of WSML formalism allows for application of a very good operational infrastructure. E.g. the currently available version of WSML2Reasoner uses the IRIS reasoner as a back-end reasoning engine. [21] reports that the functionality and performance of IRIS compare favourably with similar systems.

In this paper we have presented an organisational ontology framework for SBPM which integrates different views of an enterprise. Our ontology framework is designed with consistent level of detail and represented using a current ontology language for which scalable reasoners, repositories and tools are available. The framework has been successfully applied to a use case scenario in the telecommunication domain. By fulfilling the identified criteria to a greater extent in comparison to other approaches, the proposed organisational ontology framework has a great opportunity to become a useful and flexible tool to be applied in real world cases within different domains. With semantic annotation of business processes using organisational ontologies we also enable new types of business process verification and validation techniques.

References

1. SUPER project, http://www.ip-super.org
2. Curtis, B., Kellner, M.I., Over, J.: Process Modeling. Communications of the ACM 35(9), 75–90 (1992)
3. Jablonski, S., Bussler, C.: Workflow Management - Modeling Concepts, Architecture and Implementation. International Thomson Computer Press, London (1996)
4. Fox, M.S., Barbuceanu, M., Gruninger, M., Lin, J.: An Organisation Ontology for Enterprise Modeling. In: Prietula, M., Carley, K., Gasser, L. (eds.) Simulating Organizations: Computational Models of Institutions and Groups. AAAI/MIT Press, Menlo Park (1998)
5. Uschold, M., King, M., Moralee, S., Zorgios, Y.: The Enterprise Ontology. The Knowledge Engineer Review 13(1) (1998)
6. Fox, M.: The TOVE Project: A common-sense model of the enterprise. In: Belli, F., Radermacher, F.J. (eds.) IEA/AIE 1992. LNCS (LNAI), vol. 604, pp. 25–34. Springer, Heidelberg (1992)
7. Geerts, G., McCarthy, W.E.: An Accounting Object Infrastructure For Knowledge-Based Enterprise Models. IEEE Intelligent Systems & Their Applications (August 1999)
8. McCarthy, W.E.: The REA Accounting Model: A Generalized Framework for Accounting Systems in a Shared Data Environment. The Accounting Review (1982)
9. Lampe, J.C.: Discussion of an ontological analysis of the economic primitives of the extended-REA enterprise information architecture. International Journal of Accounting Information Systems 3, 1 (2002)
10. Gordijn, J.: E3value in a Nutshell. Technical report, HEC University Lausanne (2002)
11. SEKT Consortium. PROTON Ontology, http://proton.semanticweb.org/ (retrieved January 15, 2007), http://www.dip.semanticweb.org

12. Mühlen, M.z.: Evaluation of Workflow Management Systems Using Meta Models. In: Proceedings of the Thirty-Second Annual Hawaii international Conference on System Sciences, HICSS, vol. 5. IEEE Computer Society, Los Alamitos (1999)
13. OMG Document bmi/2006-11-02; Organization Structure Metamodel (OSM); 2nd Initial Submission; In Response to: Organization Structure Metamodel RFP (OMG Document bei/2004-06-05); Version 0.5 (November 9, 2006)
14. D.1.1. Business Process Ontology Framework, SUPER Deliverable (April 2007)
15. Hepp, M., Roman, D.: An Ontology Framework for Semantic Business Process Management Proceedings of Wirtschaftsinformatik 2007, Karlsruhe (2007)
16. Abramowicz, W., et al.: Organization Structure Description for the Needs of Semantic Business Process Management. In: Conference Proceedings of SBPM 2008 (2008)
17. Malone, Thomas, W., Sloan School of Management.: A formal model of organizational structure and its use in predicting effects of information technology, Working papers 1849-86, MIT, Sloan School of Management (1986)
18. Abramowicz, W., et al.: Semantic Enterprise Description for the Needs of Business Process Automation. In: Conference Proceedings of 1st IEEE International Workshop on Semantics for Business Process Management (SemBPM 2008) (2008)
19. DeMarco, T.: Controlling Software Projects: Management, Measurement & Estimation. Yourdon Press, New York (1982)
20. SAP Business Maps, `http://www.sap.com/solutions/businessmaps/index.epx`
21. Bishop, B., Fischer, F.: IRIS - Integrated Rule Inference System. In: Proceedings of the 1st Workshop on Advancing Reasoning on the Web: Scalability and Commonsense (ARea 2008) hosted by the 5th European Semantic Web Conference (ESWC 2008) (2008)
22. D8.1. Telecoms Vertical Domain and Process Ontology, SUPER deliverable (2007)
23. Markovic, I.: Advanced Querying and Reasoning on Business Process Models. In: Abramowicz, W., Fensel, D. (eds.) BIS 2008. LNBIP, vol. 7, pp. 189–200. Springer, Heidelberg (2008)
24. Markovic, I., Kowalkiewicz, M.: Linking Business Goals to Process Models in Semantic Business Process Modeling. In: Proceedings of the 12th IEEE International EDOC Conference, Munich, Germany (September 2008)
25. Web Service Modelling Language (WSML), WSMO Working Group Final Draft (2005)
26. Born, M., Filipowska, A., Kaczmarek, M., Markovic, I.: Business Functions Ontology and its Application in Semantic Business Process Modelling. In: Proceedings of the 19th Australasian Conference on Information Systems (2008)

A Process Oriented Assessment of the IT Infrastructure Value: A Proposal of an Ontology Based Approach

Jan vom Brocke[1], Alessio Maria Braccini[2], Christian Sonnenberg[1], and Elisabeth Ender[3]

[1] University of Liechtenstein
{jan.vom.brocke,christian.sonnenberg}@hochschule.li
[2] LUISS "Guido Carli" University
abraccini@luiss.it
[3] HILTI Befestigungstechnik AG
elisabeth.ender@hilti.com

Abstract. The impact of IT infrastructure on organizations' activities and performance is hard to evaluate since complex IT infrastructures affect multiple business processes. An integrated perspective on IT and business processes is required. Previous research dealing with process-based IT impact evaluations lacks practical applications of proposed methods. Adopting a value-based perspective, this paper focuses on the organizational impact of IT and introduces first results from an ongoing research. This paper introduces an ontology based approach to represent the relationships among business processes and IT infrastructure components.

Keywords: Ontology, Business Value, IT costs evaluation, Process Based IT Value assessment.

1 Introduction

The impact of IT infrastructure on processes and the reflexes of processes changes on IT infrastructure are hard to evaluate in a landscape where, complex IT infrastructures are linked to business processes in a network of relationships. Each IT Component can affect more than one business process [1, 2]. For example, a few years ago, the purchase department of an international manufacturing company decided to increase the frequency of the orders registration process (from weekly to daily), estimating an annual savings cost of € 400.000. However the new frequency caused an increase in the workload of the server supporting this procedure and the company had to buy a new one (cost € 450.000). The net financial performance of this operation was therefore € -50.000 for the first year. When the decision was taken, none of the two parts involved (IT and Purchase Department) were able to tell the exact effect that the modifications would have had on the IT Infrastructure.

To face problems like this, an integrative perspective of IT and business processes is required. Available research on process-based IT impact evaluations are in lack of practical applications from proposed methods [3]. The practitioners' side offers more practical approaches, providing collections of best practices adopted by organizations: both ITIL

W. Abramowicz (Ed.): BIS 2009, LNBIP 21, pp. 13–24, 2009.

v3 and CoBIT v4.1 give guidance to companies so as to manage their IT infrastructures. Even if these frameworks provide help, they fail to establish an integrated view of IT infrastructure and business processes (as described subsequently in this paper).

Another aspect that emerges in contexts like the one previously described is related to the fact that, matching IT infrastructural needs with business needs usually brings into play problems connected to shared and mutual understanding [4, 5].

This paper focuses on the organizational impact of IT management, adopting a value perspective, and introduces the first results of an ongoing research. The research question this paper is concerned with is: "By which means can IT Management be linked with the business process level?". The proposed approach uses an ontology to represent the relationships among business processes and IT infrastructure components.

The structure of this paper is as follows: after the introduction, the research methodology is described. Subsequently, the related work section serves to orientate our work within the body of available research. We then describe the proposed ontology and show its application by means of a test case. Some final remarks and considerations on further research conclude the paper.

2 Research Methodology

The research methodology of this paper follows the design science paradigm [6], aiming at the development of an ontology (the "IT artefact") that is useful to solve a problem stemming from the "business environment". The underlying design process of the ontology is displayed in Fig. 1, and is primarily based on the work of SURE et al. [7], and FOX et al. [8].

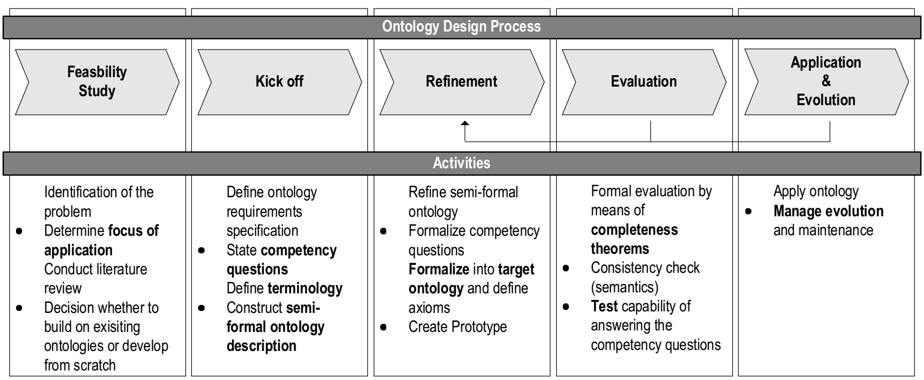


Fig. 1. Ontology Engineering Process (c.f. [7, 8])

As this paper describes a research in progress, we have not covered all phases of the design process with equal intensity. In particular, the phases of "Refinement" and "Evaluation" have been only iterated twice at the time of writing. The phase "Application and Evolution" is currently iterated the first time, so we can hardly provide resilient results from this phase for now.

Within the feasibility phase we identified the business need (practical relevance, described in the introduction), as well as the problem which was unveiled by the IT management of the company. The structure of the problem allows for a formal modeling approach and at the same time requires the models to exhibit a clear semantics. The semantic aspect is important as the modeled concepts have to be clearly understood and interpreted by all parties involved (including non-human actors). According to the design science guidelines stated by HEVNER et al. [6] we seek to apply existing foundations and methodologies from literature and practice (knowledge base) for the design of our ontology. Therefore, the concepts presented in the ontology have been strongly inspired by previous efforts of the ITIL v3 and CoBIT v4.1 frameworks as well as from preliminary work in the field of enterprise ontology.

Competency questions
Which services does IT has to offer to fulfill Business needs?
What is IT offering to the business side (service catalogue)?
What are the most infrastructure critical services?
What happens if a piece of hardware fails?
What are potential single points of failure in a given situation?
When is the IT infrastructure running into a bottleneck?
Which investments are required to solve bottlenecks?
To what extent are individual services underemployed / overburdened?
Is our IT infrastructure capable of fulfilling business requests?

Fig. 2. Sample competency questions

During the kick off phase we specified the requirements of the ontology, defining the ontology's competencies [8]. Sample competency questions are listed in Fig. 2. Within the remaining sections, we present the outcomes of the first iterations of the "Refinement" (ontology description) and the "Evaluation" (application example) phase. Prior to this, we give coordinates to position our work in the literature.

3 Related Work

Bearing in mind the proposition of this paper, the area of research which we consider to be of relevance is the one that investigates the impact of IT inside organizations under a value perspective. On this topic there is sufficient acknowledgement of the assumption that IT investments generate value inside organizations [9, 10], anyhow, the debate on how to assess or measure it is still ongoing. According to SCHEEPERS and SCHEEPERS [1] literature relevant to this topic has focused on the problem by adopting two interrelated perspectives: the IT Value and the IS Alignment perspective.

In the genesis of IT Value research the work of BRYNJOLFSSON [11] contributed to the identification of the so called "IT Paradox", intended as the absence of productivity improvements as a consequence of IT investments. Since then, IT Value has been analyzed using a wide range of different approaches [9] and theoretical perspectives [12]. The number of research papers published on this topic is high. For example, OH and PINSONNEAULT [12] and MELVILLE et al. [9], who provide extensive literature reviews on IT Value, cite (respectively) 109 and 202 papers.

Past research on IT Value was until now unable to build a consensus among researchers on the role of organizational impacts of IT investments. Even if the need to evaluate profitability and effectiveness of IT investments in organizations is a relevant priority [13], results on IT Value are not always unambiguous [14] and, moreover, they lack practical applications of proposed methodologies [3].

In a recent paper, KOHLI and GROVER [15] summarize findings about value generated by IT investments in literature with the following assumptions: IT does create value; IT creates value under certain conditions; IT-based Value manifests itself in many ways; IT-based value could be latent; there are numerous factors mediating IT and value. MELVILLE et. al [9] identified that IT investments could produce value at three different loci: the focal firm (referring to the organization investing in IT), the competitive environment and the macro environment. The difficulties connected to the identification of a proper link between IT spending and productivity induced many researchers to focus more on the focal firm. Many recent studies adopt the process perspective to analyze the value impact of IT investments [4, 2]. In particular, RAY et al. [4] highlight how IT applications tend to be process specific (effects produced at a specific process may not transfer to others), emphasizing the need for a process based IT impact evaluation.

TILLQUIST and ROGERS [16] notice that separating IT value in a process is equally difficult because it mixes with other values delivered by other resources in the process. Moreover, due to their complexity, modern IT infrastructures, may easily impact more than one process [1], creating difficulties in identifying which specific component affects a specific process or activity.

SCHEEPERS and SCHEEPERS [1], citing TALLON et al. [17] and WEILL and BROADBENT [18] identify the existence of a "dilution effect" that affects the traceability of the impact of IT investments. On the basis of this consideration they highlight the role of literature in addressing the problem of organizational value impact of IT investments under a strategic alignment and competitive environment perspective. Under this perspective the value impact of IT investments is seen as a pre-requisite for a better organizational performance that on a strategic level, can be acquired by means of a strategic fit between IT and business. According to BYRD et al. [19] literature on the strategic alignment of IS and business strategy suggests the existence of a positive effect on performance.

Perspectives adopted by research papers on IS Alignment literature are divided by SILVA et al. [20] into three main branches: managerial, emergent and critical. These approaches identify the need to adopt a managerial model so as to achieve alignment, but at the same time highlights the necessity to deal with the uncertainty of strategy formulation and the drift effect of technology [21]. The complexity of the aspects that come into play when studying the IS Alignment phenomenon makes it an ongoing or a moving target [22].

To mitigate the difficulties connected to the alignment of Strategy and IT, the business process level perspective is suggested as a vital dimension of analysis by the most cited IS Alignment framework [23] when they identify a suitable contact point between these two opposites [24, 3].

3.1 Common Contact Points and Open Issues: Enterprise Models

With the aim of identifying common traits between major trends in the areas of interest for the present work, we point out the following three elements. First of all, the process dimension of analysis can be seen as a contact point between the two perspectives adopted to investigate organizational impact of IT investments, since IT infrastructure impacts the profitability via business processes.

Consequently there is the need for a common and shared understanding between the business side and the IT side, which are involved in this context [4, 5]. Therefore, an organization might strive to adopt communication tools and mechanisms to create a shared understanding. In particular, modeling methods capable of describing the relationships between the IT Infrastructure and the business process is necessary.

Generally, Enterprise Architectures (EAs) provide the means to foster a common (model-based) understanding of an enterprise. EAs address the problem of IT and business perspective integration, i.e. IT-Business Alignment [25, 26]. In addition to IT related artifacts EAs consider business related artifacts like organizational goals, business units, products and services [26].

An enterprise model is a fundamental constituent of any EA. An enterprise model captures the entities and their relationships from multiple perspectives. Usually a hierarchical approach for modeling an "enterprise" is applied by distinguishing several architectural layers starting with a strategy or organizational layer and then establishing a hierarchy of subordinate layers (e. g. application layer, infrastructure layer). Depending on the modeling concept applied, the models may differ in their degree of formality. In particular, three generic modeling concepts can be distinguished: glossary, meta-models and ontological theories [27]. Among these modeling concepts, ontological theories exhibit the highest degree of formalization. In addition to the model concepts and their relationships (meta-model approach), ontological theories are used to comprehensively specify rules and constraints from the domain of interest [27, 28]. An ontology is commonly referred to as an explicit specification of a shared conceptualization [29, 28] and hence, can be seen as a suitable tool to create a mutual understanding among related actors. In particular, ontological theories allow for a formal analysis, execution and validation of enterprise models as well as for drawing inferences on them. Ontological theories are best suited to describe the most generic enterprise-related concepts and to define the semantics of modeling languages to be employed [27]. Due to their high degree of formalization and their capability to define semantics, ontological theories serve as an ideal means to ensure consistency of enterprise models and to reduce the number of facts to be modeled (due to the formulation of axioms and rules). Enterprise models based on an ontological theory are capable of not only answering queries of what is explicitly represented in the enterprise model (as in the traditional meta-model-based approach) but also answering queries of what is implied by the model [8], therefore allowing for a comprehensive model-based performance analysis.

Despite a considerable variety of different ontologies for individual enterprise-related phenomena, only two ontologies – the Edinburgh Enterprise Ontology (EEO, see [30]) and the TOronto Virtual Enterprise (TOVE, [31]) have been explicitly constructed for the purpose of enterprise modeling [28]. Both approaches considerably overlap in their set of concepts as they both define classes related to organizational aspects, strategy, activities and time. As opposed to EEO, TOVE has been fully translated into a target language and applied within a supply chain management scenario and thus might serve as a core ontology to be extended [28]. Both ontologies conceptualize processes, resource usage and costs. However, they do not introduce IT related concepts. Therefore we aim at conceptualizing IT artifacts and their relation to the business domain. Future research will focus on integrating the existing ontologies with our conceptualization. At the present stage, we only concentrate on conceptualizing the IT part, with the first results of this effort presented in the subsequent sections.

4 The Proposed Ontology

Fig. 3 depicts the structure of the Ontology for Linking Processes and IT infrastructure (OLPIT) indicating its classes (the grey boxes) and their relationships (the arrows). Each box contains the name of the class and its attributes. Sub-classes inherit attributes from super-classes. Inherited attributes are indicated by the (…) notation. This Ontology has been modeled using Protégé and the OWL v. 2.0 language. Due to space limitations we introduce the ontology by only using a graphical representation that aids the understanding of its constructs and their relationships, and describe its semantics in the following text.

The OLPIT builds on the ITIL v3 and COBiT v4.1 frameworks in order to consider best practices in IT management. These frameworks have been used to gain an initial understanding of the key concepts that are relevant in the problem domain and to define a first taxonomy. The taxonomy helped in the identification of a set of classes that was subsequently extended, reviewed and modified on the basis of further insights emerging from evaluation efforts.

In the proposed ontology, the Business Process is the focal point. Business processes can be understood as value interfaces as organizations deliver value to their (internal/external) customers through their business processes. Following the implications of the thought of IS Alignment, the IT infrastructure delivers value to the Business Processes via IT Services. In order to be able to reason the structural relationships between IT components and business processes as well as to reason the course of value consumption (IT cost), our ontology proposes classes and relationships of relevance in the application domain. Starting the description of the ontology from the bottom level, the IT Infrastructure is formed by IT components divided among hardware that can be Physical, Virtual or classified in Groups. A Group can be used to represent a set of hardware entities that are commonly interrelated (like for example a cluster of servers or the total ensemble of network components). In order to make the ontology schema general and not case dependant, IT Components can have extended properties associated to them (e. g. the amount of RAM, the amount of disk space, the amount of cache). IT Components, together with Human Resources, constitute the Resources that are necessary to deliver IT Services.

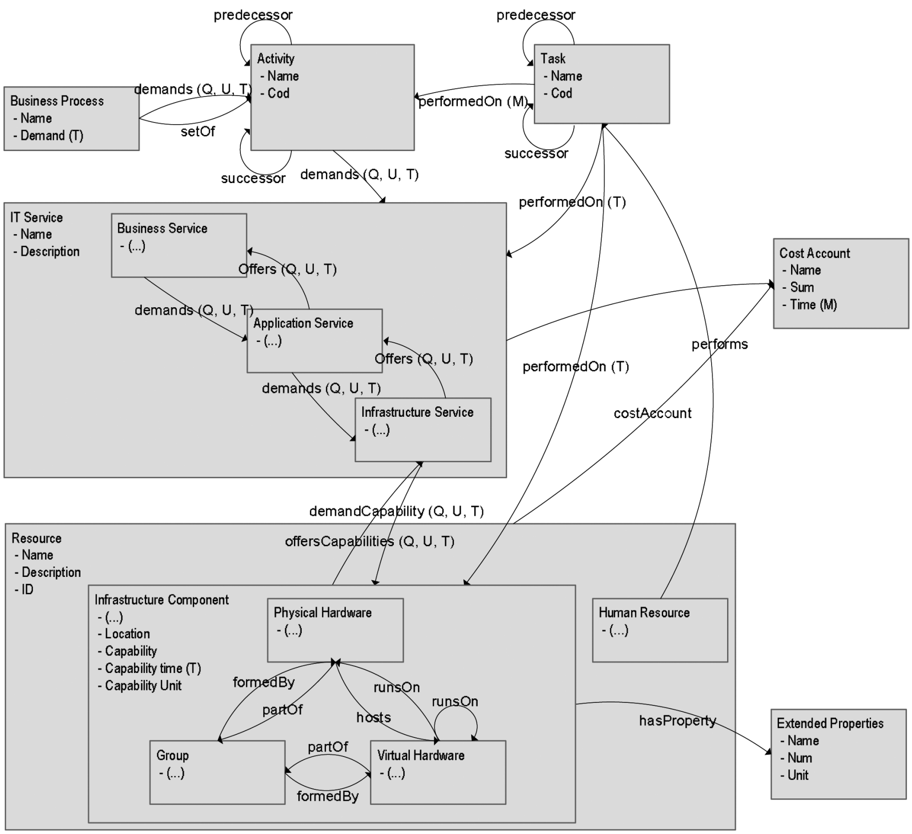


Fig. 3. The OLPIT Ontology schema

IT Services are divided into three categories: IT Infrastructure Service(s), IT Application Service(s) and IT Business Service(s). An IT Infrastructure Service delivers the capabilities of the IT Infrastructure Components to Application Services. Examples of such services could be a network service or a storage service.

An IT Application Service is a service delivered by the functions of specific software. This class is not intended to include all software (e.g. operating systems) in an IT Infrastructure, but only those which are used to deliver services to the business side. Examples of IT Application Services could be e-commerce software, content management software, ERP software and so on.

Finally, an IT Business Service, is a service that delivers value to the customer side (via Activities and Business Processes). Under this perspective, an IT Business Service contributes to the execution of one or more activities in a process. An example of IT Business Services could be a credit card verification service.

A Business Process is defined as a collection of Activities that takes inputs from other resources, manipulates them and produces outputs. Input and outputs may come from, or be directed to, other Business Process(es). An Activity may demand the execution of one (or more) IT Service(s) to deliver value or may require some Task(s) performed by Human Resource(s). Activities and tasks are linked in a chain and can have a predecessor and a successor.

The capabilities of the IT Infrastructure and the demand of the business side are represented in the ontology by means of the quantity (Q), unit (U) and time (T) constructs associated to each demand/offer relationship.

Finally, the proposed ontology models the cost information by means of the Cost Account class. A Cost Account represents a specific cost identified by its name (i.e.: depreciation), an amount (i.e.: € 1.500) and a time (i.e. year). Cost Accounts can be associated with IT Infrastructure Components, IT Services and Human Resources.

5 Test Case

In this paragraph we will introduce an example of the application of the proposed ontology in order to solve three practical problems: the measurement of the IT infrastructure capability on the base of the actual (and future) business demand, the identification of possible points of failures of the IT infrastructure and the cost calculation of a single service. The figures and the process depicted in Fig. 4 (which shows an instance of the OLPIT ontology) are not real, but act as an example.

The process indicated in Fig. 4 is a generic order entry process from an e-commerce website. In this process the customer browses the catalogue (Activity 1),

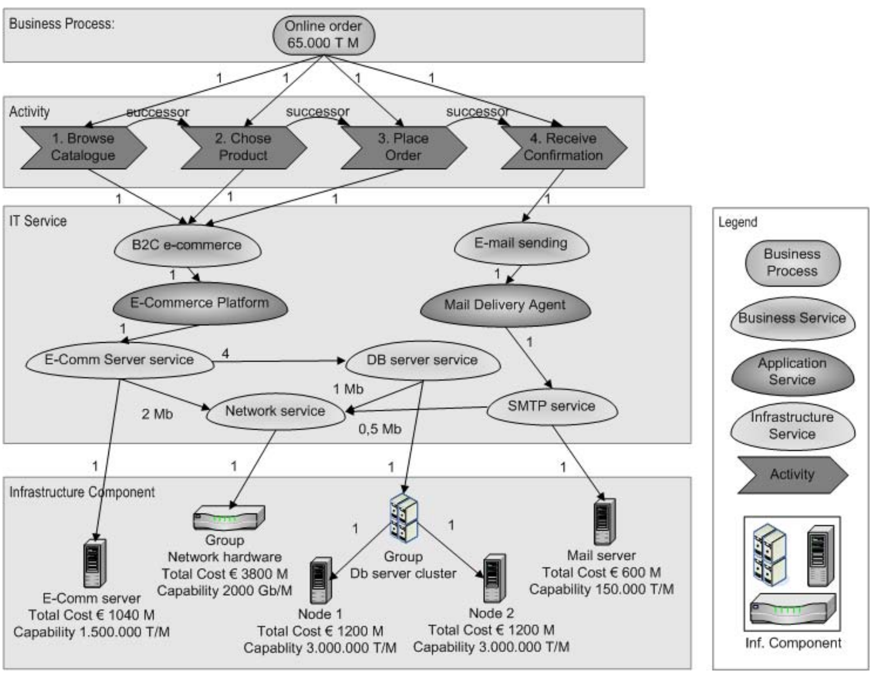


Fig. 4. A Sample Business Process modeled with OLPIT

chooses a product (Activity 2), places an order (Activity 3) and finally receives a confirmation of the order via mail (Activity 3). The business process is executed 65.000 times per month (M). To simplify costs and capability calculations, the time frame is always the month (M). Due to lack of space, costs are directly indicated as total per month per component. Each component has its own capability associated to it: the capability is indicated by a number, a unit and a time frame. For example, the capability of the e commerce server is indicated as 1.500.000 T/M, indicating that this server can offer 1,5 millions Transactions (T) per Month[1].

The numbers that are indicated in the relationships among components specify the demand: the time frame is always the month and, whenever not indicated, the unit is always "Transaction".

Fig. 4 indicates the network of relationships among all IT Components, IT Services and Activities. This network can highlight which IT Infrastructure Component affects which phase in a business process. The OLPIT ontology can be used to represent the relationship among IT infrastructure with different levels of granularity. In the test case, for example, the network has been modeled as a Group, without going too much into the details.

On the basis of the Capability and the Demand numbers presented in Fig. 4, it is possible to evaluate the balance between IT Infrastructure Capability and IT Business demand. The actual used capability of the IT infrastructure is estimated by multiplying the demand of the business side with all the factors of the relationships among the classes of the ontology and, afterwards, summing up all the demands that belong to the same Infrastructure Component. The maximum capacity of the IT Infrastructure can therefore be evaluated maximizing the function composed by the sum of all the capacity of each component multiplied by each factor: for this test case the maximum capacity of the infrastructure is close to 83.000 executions of the "Online order" process per month.

Fig. 5 (left side) shows the actual capability (C) of the infrastructure, as well as the actual request from the business side (D). If we move from a static scenario (Fig. 5 left side) to a dynamic scenario (Fig. 5 right side), and hypothesize that the total demand increases at a constant rate (0,5% each month in this example) from the point in time t_0 time it is possible to notice that the IT Infrastructure will no longer be able, ceteris paribus, to fulfill business needs at the time t_1 (49 months later).

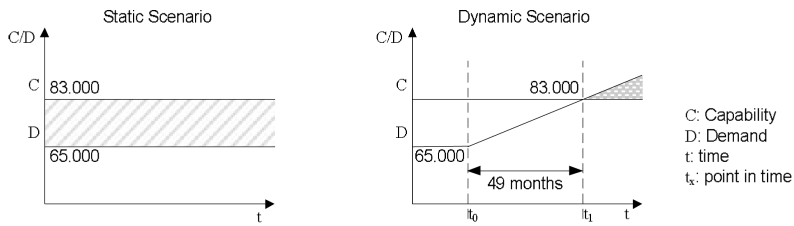


Fig. 5. IT Infrastructure capability evaluation: static and dynamic scenario

[1] In this context we use the word "Transaction" to indicate a generic request made by a service to a component: as a matter of example, a transaction for the DB server could be a query, and a transaction for the mail server could be a mail message to be sent.

Furthermore, using the capability and the demand values modeled in the ontology, possible bottlenecks can be identified. The "Usage %" column in Fig. 6 shows the actual percentage of used capacity of each IT infrastructure component. Looking at the percentages it is possible to identify the components that are about to run out of capacity (in our example, the network, with a usage percentage of 78,29%).

IT Services Cost Calculation – Service: B2C e-commerce							
Item	**Class**	**Request M**	**Capability M**	**Unit**	**Usage %**	**Total Cost M**	**Unitary Cost**
B2C e-commerce	IT Business Service	195.000		T/M		€ 7.137,30	€ 0,0355
E-Commerce Platform	IT Application Service	195.000		T/M		€ 7.137,30	€ 0,0355
E-Commerce Server Service	IT Infrastructure Service	195.000		T/M		€ 1.040,00	€ 0,0045
Database Server Service	IT Infrastructure Service	780.000		T/M		€ 2.400,00	€ 0,0020
Network Service	IT Infrastructure Service	1174,32		Gb/M		€ 3.697,30	€ 0,0038
E-Commerce Server	IT Infrastructure Component	195.000	1.500.000	T/M	13,00%	€ 1.040,00	€ 0,0045
Db Cluster	IT Infrastructure Component						
- Node 1	IT Infrastructure Component	780.000	3.000.000	T/M	26,00%	€ 1.200,00	€ 0,0010
- Node 2	IT Infrastructure Component	780.000	3.000.000	T/M	26,00%	€ 1.200,00	€ 0,0010
Network Hardware	IT Infrastructure Component	114,32	1.500	Gb/M	78,29%	€ 3.800,00	€ 0,0227
Total demand (M)	65.000						

Fig. 6. IT Service total cost calculation

Finally, the cost information modeled with the ontology enables the reconstruction of the total cost of each service. By means of an example Fig. 6 shows costs associated with each component. The total cost of the services is the total sum of all the costs of the components that belong to it. For shared services (like the network in our case), the cost is divided on the base of the total usage. In the example, the total cost of the Network Hardware is equal to € 3.800 but the actual cost of the Network Service (which will be part of the total cost of the B2C e-commerce service) is only € 3.697,30 (a quota calculated on the basis of the amount of network traffic generated by this service). The other quota forms the total cost of the "E-Mail sending" business service, not covered in this example.

6 Conclusion

In this paper we focused on the organizational impact of IT management adopting a value perspective. As a first result of an ongoing research we introduce an ontology that links IT management to the business process level. The proposed ontology is based on a lean set of classes and, according to our opinion, forms a good base for future extensions. The testing section of this paper demonstrates practical problems that can be solved by means of our ontology.

A current limitation of our approach is the limited number of test iterations our ontology has undergone. Since this is the first result of an ongoing research, our ontology has been tested with experimental data only, and only a small part of them is currently used in the real context.

A further limitation deals with the value perspective adopted, because we are only addressing the passive side of the financial cycle (only costs). Nevertheless, since we decided to adopt an ontology based approach due to the possibility of integration of different ontologies, we can now think about integrating our ontology with others available and in doing so address the positive side of the financial cycle. For these reasons, future research will be addressed to fully test the proposed ontology in a real context and to evaluate its possible integrations with existing (and related) ontologies for enterprise modeling.

References

1. Scheepers, H., Scheepers, R.: A process-focused decision framework for analyzing the business value potential of IT investments. Information Systems Frontiers 10(3), 321–330 (2008)
2. Tallon, P.P.: Does IT pay to focus? An analysis of IT business value under single and multi-focused business strategies. Journal of Strategic Information Systems 16(3), 278–300 (2007)
3. Leem, C.S., Yoon, C.Y., Park, S.K.: A Process-Centered IT ROI Analysis with a Case Study. Information Systems Frontiers 6(4), 369–383 (2004)
4. Ray, G., Muhanna, W.A., Barney, J.B.: The Role of Shared IT-Business Understanding. Communications of the ACM 50(12), 87–91 (2007)
5. Luftman, J.N., Kempaiah, R.: An Update on Business-IT Alignment: A line Has Been Drawn. MIS Quarterly Executive 6(3), 165–177 (2007)
6. Hevner, A.R., March, S.T., Park, J., Ram, S.: Design Science in Information Systems Research. MISQ 28(1), 75–105 (2004)
7. Sure, Y., Akkermans, H., Broekstra, J., Davies, J., Ding, Y., Duke, A., Engels, R., Fensel, D., Horrocks, I., Iosif, V., Kampman, A., Kiryakov, A., Klein, M., Lau, T., Ognyanov, D., Reimer, U., Simov, K., Studer, R., van der Meer, J., van Harmelen, F.: On-to-knowledge: semantic web enabled knowledge management. In: Zhong, N., Liu, J., Yao, Y. (eds.) Web Intelligence, pp. 277–300. Springer, Berlin (2003)
8. Fox, M.S., Barbuceanu, M., Gruninger, M., Lin, J.: An Organization Ontology for Enterprise Modelin. In: Prietula, M., Carley, K., Gasser, L. (eds.) Simulating Organizations: Computational Models of Institutions and Groups, p. 131. AAAI/MIT Press, Menlo Park (1998)
9. Melville, N., Kraemer, K., Gurbaxani, V.: Review: Information technology and organizational performance: an integrative model of IT business value. MIS Quarterly 28(2), 283–322 (2004)
10. Kohli, R., Sherer, S.A., Baron, A.: Editorial – IT Investment Payoff in E-Business Environments: Research Issues. Information Systems Frontiers 5(3), 239–247 (2003)
11. Brynjolfsson, E.: The contribution of information technology to consumer welfare. Information Systems Research 7(3), 281–300 (1993)
12. Oh, W., Pinsonneault, A.: On the assessment of the strategic value of information technologies: conceptual and analytical approaches. MIS Quarterly 31(2), 239–265 (2007)
13. Seddon, P.B., Graeser, V., Willcocks, L.P.: Measuring Organizational IS Effectiveness: an Overview and Update of Senior Management Perspectives. The DATA BASE for Advances in Information Systems 33(2), 11–28 (2002)
14. Im, K.S., Dow, K.E., Grover, V.: Research Report: A Reexamination of IT Investment and the Market Value of the Firm – An Event Study Methodology. Information Systems Research 12(1), 103–117 (2001)

15. Kohli, R., Grover, V.: Business Value of IT: An Essay on Expanding Research Directions to Keep up with the Times. Journal of the Association for Information Systems 9(1), 23–39 (2008)
16. Tillquist, J., Rodgers, W.: Using Asset Specificity and Asset Scope to Measure the Value of IT. Communications of the ACM 48(1), 75–80 (2005)
17. Tallon, P., Kraemer, K., Gurbaxani, V.: Executives' perceptions of the business value of information technology: A process-oriented approach. Journal of Management Information Systems 16(4), 145–173 (2000)
18. Weill, P., Broadbent, M.: Leveraging the new infrastructure. Harvard Business School Press, Boston (1998)
19. Byrd, T.A., Lewis, B.R., Bryan, R.W.: The leveraging influence of strategic alignment on IT investment: an empirical examination. Information & Management 43(3), 308–321 (2006)
20. Sylva, L., Figueroa, E.B., Gonzalez-Reinhart, J.: Interpreting IS alignment: a multiple case study in professional organizations. Information and Organization 17(4), 232–265 (2007)
21. Ciborra, C.U.: The labyrinths of information: challenging the wisdom of systems. Oxford University Press, Oxford (2002)
22. Chen, R.S., Sun, C.M., Helms, M.M., Jih, W.K.: Aligning Information Technology and business strategy with a dynamic capabilities perspective: a longitudinal study of a Taiwanese semiconductor company. International Journal of Information Management 28(5), 366–378 (2008)
23. Henderson, J.C., Venkatraman, N.: Strategic alignment: leveraging information technology for transforming organizations. IBM Systems Journal 32(1) (1993)
24. Tallon, P., Kraemer, K.: A process-oriented assessment of the alignment of information systems and business strategy: implication for IT business value. In: Proceedings of the 4th Americas Conference on Information Systems, Baltimore (1999)
25. Lankhorst, M.M.: Enterprise architecture modelling – the issue of integration. Advanced Engineering Informatics 18(4), 205–216 (2003)
26. Winter, R., Fischer, R.: Essential Layers, Artifacts, and Dependencies of Enterprise Architecture. Journal of Enterprise Architecture 3(2), 7–18 (2007)
27. IFIF-IFAC Task Force: GERAM: The Generalised Enterprise Reference Architecture and Methodology, Version 1.6.3. In: Bernus, P., Nemes, L., Schmidt, G. (eds.) Handbook on Enterprise Architecture, p. 21. Springer, Berlin (2003)
28. Grünninger, M.: Enterprise Modelling. In: Bernus, P., Nemes, L., Schmidt, G. (eds.) Handbook on Enterprise Architecture, p. 515. Springer, Berlin (2003)
29. Gruber, T.R.: A translation approach to portable ontology specifications. Knowledge Acquisition 5(2), 199–220 (1993)
30. Uschold, M., King, M., Moralee, S., Zorgios, Y.: The Enterprise Ontology. Knowledge Engineering Review (1997)
31. Grünninger, M., Fox, M.S.: Enterprise Modelling. AI Magazine 19(3), 109–121 (1998)

A Semantic Clinical Knowledge Representation Framework for Effective Health Care Risk Management

Ermelinda Oro[1], Massimo Ruffolo[2], and Domenico Sacca[1]

[1] Department of Computer Science and System Science (DEIS)
{linda.oro,sacca}@deis.unical.it
[2] Institute of High Performance Computing and Networking of CNR (ICAR-CNR),
University of Calabria, 87036 Rende (CS), Italy
ruffolo@icar.cnr.it

Abstract. A process-oriented vision of clinical practices may allow to enhance patient safety by enabling better risks management capabilities. In the field of clinical information systems less attention has been paid to approaches aimed at reducing risks and errors by integrating both declarative and procedural aspects of medical knowledge. This work describes a semantic clinical knowledge representation framework that allows representing and managing, in a unified way, both medical knowledge and clinical processes and supports execution of clinical process taking into account risk handling. Framework features are presented by using a running example inspired by the clinical process adopted for caring breast neoplasm in the oncological ward of an Italian hospital. The example shows how the proposed framework can contribute to reduce risks.

Keywords: Semantic Business Process Management, Ontology, Knowledge Representation and Reasoning, Workflow Representation, Logic Programming, Health Care Information Systems.

1 Introduction

Nowadays health care risks management is a high priority theme for health care professionals and providers. Across the world the whole issue of patient safety, medical errors prevention and adverse events reporting is a very challenging and widely studied research and development topic that stimulates a growing interest in the computer science researchers community. For reducing risks and enhancing patient safety, a promising approach is the definition of process-oriented clinical information systems. In fact, health care services and practices are characterized by complex clinical processes in which high cost and risk activities take place. A clinical process can be seen as a workflow where clinical (e.g. treatments, drugs administration, guidelines execution, medical examinations, risk evaluation) and general (e.g. patient enrolment, medical record instantiation) activities and events occur. Clinical processes and their activities are, also, characterized by specific and sophisticated medical knowledge. Systems that provide integrated functionalities for: (i) representing and managing medical knowledge with related risks and errors conditions; (ii) executing clinical processes and monitoring possible risks and errors, can change clinical practices and help diffusion of a process and

W. Abramowicz (Ed.): BIS 2009, LNBIP 21, pp. 25–36, 2009.

quality awareness in health care organizations. A large body of work is currently available in the field of medical knowledge and clinical processes representation and management. Very famous approaches and tools for medical knowledge representation are: CCAM [1], CPT [2], ICD [3,4], LOINC [5], OPCS4 [6], SNOMED CT [7], MeSH [8] and UMLS [9]. In the clinical process field the evidence-based medicine movement, that aims at provide standardized clinical guidelines for treating diseases [10], has stimulated the definition of a wide set of approaches, languages and tools for representing and managing guidelines. Well known formalisms are GLIF [11], Pro*forma* [12], Arden Syntax [13,14], EON [15] based on different paradigms. For instance, Pro*forma* is a process description language grounded in a logical model, whereas GLIF is a specification consisting of an object-oriented model. Systems that provide comprehensive framework for managing clinical guidelines are DeGeL [16] and SEBASTIAN [17]. Existing systems and approaches suffer of the following shortcomings: (i) they have a lack of mechanisms for errors and risks handling and prevention; (ii) they do not use the same formalism for representing and managing both medical knowledge and clinical processes, hence, they are not able to exploit in a fully unified way declarative and procedural knowledge during execution and monitoring of clinical process; (iii) they do not allow to organize clinical processes and their element as ontologies; (iv) they do not allow to modify and customize represented knowledge and to execute clinical process in a flexible and agile way. This work describes an Ontology-based Clinical Knowledge Representation Framework (OCKRF), that aims at supporting a semantic process-centered vision of health care practices, and its prototypical implementation. The main contribution of the paper consists in the definition of an ontology-based approach to medical knowledge representation that allows to express in a combined way medical ontologies, clinical processes and related risks and errors rules. More in detail, the OCKRF allows methods for: (i) creating ontologies of clinical processes that can be queried and explored in a semantic fashion; (ii) expressing errors and risks rules (by means of *reasoning tasks*) that can be used (during processes execution) for monitoring processes; (iii) executing clinical processes and acquiring clinical process instances by means of either *workflow enactment* (predefined process schemas are automatically executed) or *workflow composition* (activity to execute are chosen step-by-step by humans); (iv) monitoring clinical processes during the execution by running reasoning tasks; (v) analyzing acquired clinical process instances, by means of querying and inference capabilities, in order to recognize errors and risks for patients. The OCKRF allows representing and managing both static and dynamic aspects of medical knowledge. It enables to adopt semantics for better designing, executing, controlling and managing clinical processes. The remainder of this paper is organized as follows. Section 2 briefly shows the process meta-model adopted in the OCKRF and sketches a running example. Section 3 presents theoretical foundations of OCKRF and shows how to use it by example. Section 4 describes the OCKRF prototypical implementation. Finally, Section 5 concludes the paper and draws future work.

2 Process Modelling

A significant amount of research has been already done in the specification of mechanisms for process modeling (see, [18] for an overview of different proposals). The most widely adopted formalism is the control flow graph, in which a workflow is represented by a labeled directed graph whose nodes correspond to the activities to be performed, and whose arcs describe the precedences among them. In this paper, we adopt the graph-oriented workflow meta-model shown in Figure 1.a and 1.b, inspired by the JPDL [19] process modeling approach. The adopted meta-model: (i) covers the most important and typical constructs required in workflow specification; (ii) allows to implement the OCKRF by using the JBPM workflow engine; (iii) allows to use workflow mining techniques grounded on graph-oriented meta-models. Since our scope is to extend the meta-model by semantic features (as described in Section 3) we need firstly to formally define it as the following 6-tuple: $\mathcal{P} = \langle N, A_r, E_v, A_n, T_k, E \rangle$, where: (i) N is a finite set of *nodes* partitioned in the following subsets: task nodes N_T (that represent activities in which a human perform tasks), subprocess nodes N_{SP} (that model activities that refer processes external to the current one), group nodes N_G (that represent a set of nodes that can be executed without a specific order), custom nodes N_C (that model activities in which custom methods can be executed and handled automatically), wait nodes N_W (that represent activities that temporary stop the execution while they execute methods), join nodes N_J , fork nodes N_F (that are respectively used to combine or split execution paths) and decision nodes N_D (that allow to control the execution flow on the base of conditions, variables or choices performed by human actors). (ii) A_r is a set of *actors*. Actors can be human or automatic. They represent the agents that execute a given task or activity. (iii) A_n is a set of *actions*. An action is a special activity that can be performed as answer to the occurrence of an event. (iv) T_k is a set of *tasks* that represent tasks to execute in task nodes. (v) $E = \{\langle x, y \rangle : x \in N_{From} \wedge y \in N_{To}\}$ is a set of *transitions* in which the following restrictions hold, when $N_{From} \equiv N_{FN} \cup N_D$ then $N_{To} \equiv N_{FN} \cup N_{FCN} \cup N_{end}$ and when $N_{From} \equiv N_{FCN}$ then $N_{To} \equiv N_{FN} \cup N_D$. Moreover, for each process there is a transition of the form $e_{start} = \langle N_{start}, y \rangle$ where $y \in N_{FN}$ and one of the form $e_{end} = \langle x, N_{end} \rangle$ where $x \in \{N_{FN} \cup N_D\}$. The subset $E_d \subset E$ where $E_d = \{\langle x, y \rangle : x \in N_D \wedge y \in N_{FN}\}$ is the set of *decisions*. A decision relates a decision node to a flow node and could hold a *decision rule* that is used at run-time to automatically control the execution flow of a process. (vi) E_v is a set of *events*. An event causes the execution of an action that constitutes the answer to the event. An event can be, for example, the throwing of an exception during the execution of a task.

2.1 A Clinical Process for Caring Breast Neoplasm

This section describes a clinical process for caring the breast neoplasm (Figure 1.c). This process will be used in the rest of the paper as running example for demonstrating OCKRF features. The example considers practices carried out in the oncological ward of an Italian hospital, hence it is not general but specific for the domain of the considered ward. The clinical process is organized in the following 10 activities: (i) Task node *Acceptance* models patient enrollment. A patient arrives to the ward with an already existing clinical diagnosis of a breast neoplasm. The activity is manually executed by an oncologist that collects patient personal data. (ii) Group node *Anamnesis* represents a set of anamnesis activities: *general anamnesis* in which physiological general data (e.g. allergies, intolerances) are being collected; *remote pathological anamnesis*, concerning past pathologies; *recent pathological anamnesis*, in which each data or

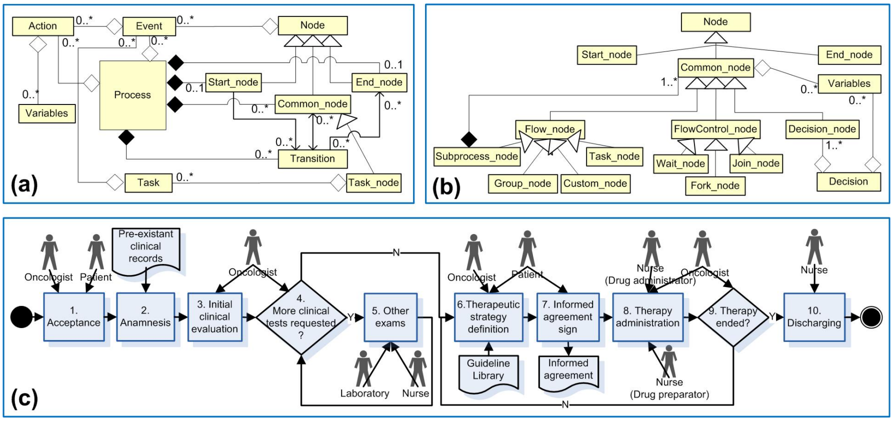


Fig. 1. (a) The process meta-model. (b) The nodes hierarchy. (c) A clinical process for caring the breast neoplasm.

result derived from examinations concerning the current pathology (or pathologies) are acquired. These activities can be executed without a specific order. (iii) Task node *Initial clinical evaluation* allows to acquire the result of an examination of the patient by an oncologist. (iv) Decision node *More clinical test requested* represents the decision to perform or not additional examination on the patient. (v) Group node *Other exams* models possible additional clinical tests. If requested these tests are conducted to find out general or particular conditions of patient and disease not fully deducible from the test results already available. (vi) Task node *Therapeutic strategy definition* models the selection of a guideline with related drug prescription. At execution time the physician picks a guideline (selected among the guidelines already available in the knowledge base) that depends upon actual pathology state as well as other collected patient data. (vii) Task node *Informed agreement sign* models the agreement of the patient concerning understanding and acceptance of consequences (either side effects or benefits) which may derive from the chosen chemotherapy, and privacy agreements. (viii) Sub-process *Therapy administration*, models a subprocess that constitutes the guideline to execute for caring the patient. (ix) Decision node *Therapy ended* models a decision activity about effects of the therapy and the possibility to stop or continue cares. (x) Task node *Discharging* models the discharging of the patient from the ward end allows to acquire final clinical parameter values. In the activities (vi) and (viii) risk and error conditions can be identified. At each guideline, chosen in (vi), corresponds a prescription of drugs (chemotherapy). Hence the computation of doses, which may depend on patient's biomedical parameters such as body's weight or skin's surface, is required. Cross-checking doses is fundamental here, because if a wrong dose is given to the patient the outcome could be lethal. Furthermore, therapy administration ((viii)-th activity) must contain checks that aims at verify type and quantity of chemotherapeutic drugs to submit to the cared patient.

3 Ontology-Based Clinical Knowledge Representation Framework

The Ontology-based Clinical Knowledge Representation Framework (OCKRF) is based on the DLP+ ontology representation language [20] that, beside complete and expressive ontology representation features, holds also powerful ASP reasoning capabilities [21,22] over represented knowledge. Thus the OCKRF allows to represent in a combined way clinical processes (procedural medical knowledge) and medical ontologies (declarative medical knowledge). By means of user friendly interfaces the framework is well suited for enabling agile knowledge representation. Furthermore, it provides powerful reasoning mechanisms that works over represented knowledge. OCKRF reasoning capabilities allow to express decisions, risk and error rules in a declarative way. More in detail, the OCKRF allows to represent extensional and intensional aspects of both declarative and procedural medical knowledge by means of: (i) *Medical Ontology and Clinical Process Schemas*. The former expresses concepts related to different medical domains (e.g. diseases, drugs, medical examinations, medical treatments, laboratory terms, anatomy, patients administration, risks). Ontology contents can be obtained by importing other existing medical ontologies and thesaurus or by means of direct manual definition. The latter are expressed according with the workflow meta-model illustrated in Section 2. The key idea which the framework is based on is that elements of the workflow meta-model (i.e. processes, nodes, tasks, events, transitions, actions, decisions) are expressed as ontology classes. This way workflow elements and medical knowledge can be easily combined in order to organize clinical processes and their elements as an ontology; (ii) *Ontology and Process Instances* both expressed in term of ontology instances. In particular, ontology class instances can be obtained by importing them from already existing medical ontologies or by creating them during process execution. Clinical process instances are created exclusively during process execution. Instances are stored in a knowledge base; (iii) *Reasoning Tasks* that express decision and risk rules computed by exploiting reasoning capabilities of DLP+ [20,22]. More formally an ontology in the OCKRF is the 5-tuple: $O = \langle D, A, C, R, I \rangle$. In the following the meaning of O and the DLP+ language syntax (that express them) are explained by describing the implementation of the running example presented in Section 2.1.

3.1 Medical Ontology and Clinical Process Schemas

Schemas are expressed by using elements of D, A, C and R in O_Z that are *finite* and *disjoint* sets of entity names respectively called *data-types*, *attribute-names*, *classes* and *relations*. The set of classes C is organized in taxonomies and partitioned in two subsets: (i) the set of process classes $C_P = N \cup A_r \cup A_n \cup T_k \cup E_v$ that represents elements of the workflow meta-model. It is constituted by the union of classes representing nodes, actors, actions, tasks and events; (ii) the set of medical classes C_M that represent concepts related to different medical domains. The set R is partitioned in two subsets: (i) the set of relations $R_P = E \cup D_c$ aimed at representing transitions and decisions; (ii) the set of relations R_M used for representing medical ontologies.

A class can be thought as an aggregation of individuals (objects) that have the same set of properties (attributes). From a syntactical point of view, a class is a name and an ordered list of attributes identifying the properties of its instances. Each attribute is identified by a name and has a type specified as a data-type or class. In the following the DLP+ implementation of the workflow meta-model is presented. In particular, *nodes* in C_P are implemented by using the class hierarchy (built up by using `isa` key-word) shown below.

```
class process(name:string).
class node(name:string, container: process, start_time:integer, end_time:integer).
    class start_node() isa{node}.
    class end_node() isa{node}.
    class common_node () isa{node}.
        class flowControl_node() isa{common_node}.
            class fork() isa{flowControl_node}.
            class join() isa{flowControl_node}.
            class wait_node() isa{flowControl_node}.
        class flow_node() isa{common_node}.
            class task_node(tasks:[task], handler:human_actor) isa{flow_node}.
            class custom_node(handler: automatic_actor, method:string) isa{flow_node}.
            class group_node(nodes:[node]) isa{flow_node}.
            class sub_process_node(sub_proc: process) isa{flow_node}.
        class decision_node(handler:actor) isa{common_node}.
            class automatic_decision_node(handler:automatic_actor) isa{decision_node}.
            class manual_decision_node(task:task, handler:human_actor) isa{decision_node}.
```

Task nodes and manual decision nodes contain *tasks* that are performed by humans. Tasks (`class task(name: string).`) collects values of activity variables given in input by human actor. *Actors* of a process (that can be human or automatic) represent the agents that execute a given task. They are represented by means of the following classes in C_P:

```
class actor(name:string).
    class human_actor() isa {actor}.
    class automatic_actor(uri:string) isa {actor}.
```

During the process enactment, by running risk and errors rules, *events* may occur. Furthermore, an event can be generated by an exception during the execution of a task. Events, and related actions to performs in response, are represented in C_P by the following classes.

```
class event(relativeTo:object, timestamp:integer).
    class node_event(relativeTo:node) isa{event}.
    class task_event(relativeTo:task) isa{event}.
    class process_event(relativeTo:process) isa{event}.
class action(method:string).
```

Relationships among objects are represented by means of relations, which like classes, are defined by a name and a list of attributes. *Transitions* and *decisions*, in R_P, that relate couple of nodes, are represented by means of the following ontology relations.

```
relation transition(name:string, from:node, to:node).
relation decision(name:string, from:decision_node, to:node).
```

When the user defines a specific process schema s/he can specialize original meta-model elements for adding new semantic attribute required by the specific process. In the following are shown some classes representing nodes of the running example depicted in section 2.1.

```
class acceptance_node(tasks:[acceptance_form],handler:physician) isa{task_node}.
class anamnesis_node(nodes:[general_anamnesis_node,remotePathological_anamnesis_node,
      recentPathological_anamnesis_node]) isa {group_node}.
class recentPathological_anamnesis_node(tasks:[pathology_form],handler:physician)
      isa {task_node}.
class therapeutic_strategy_definition_node(tasks:[therapeutic_strategy_form],handler:nurse)
      isa {task_node}.
class therapy_administration_node(sub_process:therapy_administration_process)
      isa{sub_process_node}.
class more_tests_node(task:more_tests_form) isa{manual_decision_node}.
```

`acceptance` and `therapeutic_trategy_definition` process activities are represented as subclasses of `task_node` class, in fact they represent activities in which tasks consist in the execution of forms filled by humans. Whereas `anamnesis_node`, which

Recent Pathological anamnesis activity belongs to, is represented as a subclass of group_node class. therapy_administration_node and more_tests_node are specializations of sub_proc_node and decision_node respectively. Human actors that operate in the clinical process used as running examples could be physicians, nurses and patients. They are represented by a person hierarchy that exploits multiple inheritance capabilities of DLP+ in order to express that persons are also human actors of the clinical process.

```
class person(fiscalCode:string,name:string,surname:string,sex:sex_type,
             bornDate:date,address:address).
    class patient(hospitalCard:string, weight:float, heigthCm:float) isa {person,human_actor}.
    class healthCareEmploy(occupation:string, role:string) isa {person,human_actor}.
        class nurse() isa {healthCareEmploy}.
        class physician() isa {healthCareEmploy}.
```

Class schemas representing tasks related to task-nodes can be expressed by using the following class schemas. Attribute types can be classes represented in C_M expressing different medical concepts (e.g. diseases, drugs, body parts). During task execution values of resulting class instances are obtained from fields filled in forms.

```
class task(name: string).
    class acceptance_form(patient:patient, acc_date:date) isa{task}.
    class pathology_form(disease:disease) isa{task}.
    class chemotherapeutic_strategy_form(strategy:therapeuticStrategy) isa{task}.
    class more_tests_form(choice:boolean)isa{task}.
```

In a clinical process, an event can be activated by an exception during the execution of a node or by a reasoning task aimed at control possible risks and errors. A reasoning task checks parameters values of running node and already acquired node instances and throws an event related to an error. An example of different kinds of possible errors is shown in the following taxonomy, where the attribute msg of the class view_msg (action) is the message to display when the error occurs.

```
class task_event(relativeTo:task) isa{event}.
    class medicalError(msg:string) isa{task_event}.
        class drugPrescriptionError() isa {medicalError}.
class view_msg(msg:string) isa {action}.
```

Class schemas in C_M expressing knowledge concerning anatomy, breast neoplasm disease and related therapies and drugs have been obtained (imported) from the Medical Subject Headings (Mesh) Tree Structures [8], the International Classification of Diseases (ICD10-CM) [23] and the Anatomical Therapeutic Chemical (ATC/DDD) [24].

```
class anatomy(name:string).
    class bodyRegion() isa {anatomy}.
class disease(descr:string).
    class neoplasm() isa {disease}.
        class malignant_neoplasm() isa {neoplasm}.
            class primarySited_neoplasm(site:bodyRegion,zone:string) isa {malignantNeoplasm}.
                class breast_primarySited_neoplasm() isa {primarySited_neoplasm}.
class drug(name:string, ddd:float, unit:unitOfMeasure,admRoute:[string], notes:string).
    class antineoplasticAndImmunomodulatingAgent() isa {drug}.
        class endocrineTherapy() isa {antineoplasticAndImmunomodulatingAgent}.
            class hormoneAntagonistsAndRelatedAgents()isa {endocrineTherapy}.
                class enzymeInhibitors()isa {hormoneAntagonistsAndRelatedAgents}.
            class hormoneAndRelatedAgents()isa {endocrineTherapy}.
                class estrogens() isa {hormoneAndRelatedAgents}.
class code(c:string).
    class icd10Code(chapter:integer, block:string,category:string, subCat:string) isa {code}.
    class mesh08Code(category:string, subCat:string) isa {code}.
class therapy(name:string, dru:drug, dose:float).
class therapeuticStrategy(patient:patient, therapy:therapy,startDate:date, nDay:integer).
```

The previous classes are a fragment of a medical ontology inherent (breast) neoplasm cares and are used to model the clinical process shown in Section 2.1. Class primarySited_neoplasm shows the ability to specify user-defined classes as attribute types (i.e. site:bodyRegion). Class drug has a list-type attribute admRoute:[string] representing possible Route of administration for a drug (for example inhalation, nasal, oral, parenteral). Relation schemas expressing medical knowledge can be declared by using the following syntax:

```
relation suffers (patient:patient, disease:disease).
relation relatedDrug (dis:disease, dru:drug).
relation sideEffect (dru:drug, effect:string).
relation classifiedAs (dis:disease, c:code).
```

Relation suffer asserts diseases suffered by a patient. Relations relatedDrug and sideEffect associates respectively drugs to a diseases and side effects to drugs. Moreover, relation classifiedAs enables users to query the ontologies by using codes defined in the original medical ontologies.

3.2 Ontology and Process Instances

Clinical process instances are expressed by ontology instances and created exclusively during process execution. Classes instances (objects) are defined by their *oid* (that starts with #) and a list of attributes. Instances obtained by executing the running example, are shown in the following.

```
#1:neoplasm_process(name:"Breast Neoplasm").
#2:therapy_administration_process(name:"Therapy Administration").
#1_1:acceptance_node(name:"Acceptance", container:#1, start_time:6580, end_time:16580,
     tasks:[#1_1_1], handler:#27).
#1_2:anamnesis_node(name:"Anamnesis", container:#1, start_time:16570, end_time:26580,
     nodes:[#1_2_1, #1_2_2, #1_2_3])
#1_2_3:recentPathological_anamnesis_node(name:"Recent Pathological Anamnesis", container:#1,
       start_time:19580, end_time:26570,tasks:[#1_2_3_1],handler:#27).
...
```

As described in section 2.1, instance of anamnesis_node #1_2 is composed by a set of anamnesis activities represented by means of their *id*. The object #1_2_3 belongs to #1_2. Objects #1_1, #1_2_3 are tasks executed in custom and manual decision node and are stored as their attributes. When execution arrives in a task node or in a manual decision node, task instances is created and the user input is stored as values of the task attributes. Some tasks related to task nodes are shown in the following.

```
#1_1_1:acceptance_form(name:"Acceptance", patient:#21, acc_date:#data_089).
#1_2_3_1:pathology_form(name:"Recent Pathology", disease:#neoB_01).
```

For example, acceptance_form and object is obtained by a form filled by an oncologist. It contains an instance of patient class.

Transition and decision tuples, created during the process execution, are shown in the following. In the example, the decision is obtained as a manual choice of an oncologist, but instances of decisions could be automatically generated by means of reasoning tasks.

```
transition(name:"Acceptance-Anamnesis",from:#1_0, to:#1_1).
decision(name:"More Clinical Tests requested - No",from:#1_4, to:#1_6).
```

By considering the running example, instances for the classes bodyRegion, breast_primarySited_neoplasm, for the subclasses of drug and code, can be obtained by importing them from already existing medical ontologies and can be declared as follows:

```
#A01.236: bodyRegion(name:"breast").
#neoB_01: breast_primarySited_neoplasm(descr:"Malignant neoplasm of breast", site:#A01.236,
          zone:"Nipple and areola").
```

```
#L02BG03: enzymeInhibitors(name:"Anastrozole", ddd:1, unit:mg, admRoute:["oral"], notes:"").
#L02AA04: estrogens(name:"Fosfestrol", ddd:0.25, unit:g, admRoute:["oral","parenteral"],
                    notes:"").
#icd10_C50.0: icd10Code(c:"C50.0", chapter:2, block:"C", category:"50", subCat:"0").
#mesh08_C04.588.180: mesh08Code(c:"C04.588.180",category:"C", subCat:"04").
```

The object having ***id*** `#neoB_01`, is an instance of the `breast_primarySited_neoplasm` class. Its attributes `descr` and `zone` (which type is `string`) have respectively value `"Malignant neoplasm of breast"` and `"Nipple and areola"`, whereas the attribute `site` has value `#A01.236` that is an ***id*** representing an instance of the class `bodyRegion`. Tuples expressing medical knowledge can be declared by using the following syntax:

```
suffer (pat:#21, dis:@neoB_01).
relatedDrug (dis:@C50.9, dru:@L02BG03).
sideEffect (dru:@L02BG03, effect:"Chest pain").
sideEffect (dru:@L02BG03, effect:"Shortness of breath").
classifiedAs (dis:@neoB 01, c:@icd10 C50.0).
classifiedAs (dis:@neoB 01, c:@mesh08 C04.588.180).
```

The tuple of the relation `suffer` asserts that the patient `@p_002` suffers of the disease `@neoB_01`. The same diseases is classified in the ICD10-CM with identifier code `@icd10_C50.0`, and is stored in Mesh tree structure with identifier code @mesh08_C04.588.180. By means of the relation `classifiedAs` an user is enabled to querying ontology concept referring to the correspondent identifiers.

3.3 Reasoning over Schemas and Instances

Since the OCKRF is built on top of DLP+ [22, 20], integrity constraints and complex inference rules can be expressed over schemas and instances respectively by means of *axioms* and *reasoning tasks*. For example, the following axiom prevents the prescription of a drug to a patient that has an allergy to a particular constituent of the drug.

```
::-therapyStrategy(patient:P, therapy:T, drug:D),hasActivePrinciple(drug:D,constituent:C),
   allergy(patient:P,actPrin:C).
```

Axioms could be, also, used for: (i) specify constraints about transitions behavior. For example, the axiom `"::-P:process(), not start_node(container:P)."` expresses that a `start_node` must exist for each process. Constraints express, also, that a transition links nodes belonging to the same process, and corresponds to an effective edge of the process model as shown in the following:

```
::-transition(from:N1,to:N2), N1:node(container:P1),   N2:node(container:P2), P1!=P2.
::-transition(from:N1,to:N2), N1:node(start_time:ST1), N2:node(start_time:ST2), ST1>=ST2.
::-P:neoplasm_process(), transition(from:N1,to:N2),    N1:acceptance_node(container:P),
   not N2:anamnesis_node(container:P).
...
```

A reasoning task can be used to throw a medical error when the prescribed dose exceed the recommended dose based on individual characteristics (i.e. age and weight) of the interested patient. Such a check is useful when a `therapeutic_strategy_form` is created while `therapeutic_strategy_definition_node` is active.

```
ID:drugPrescription_medicalError(relativeTo:TASK,timestamp:TIME,msg:MSG):-
  TASK:chemotherapeutic_strategy_form(strategy:STR),STR:therapeuticStrategy(patient:P,therapy:T),
  P:patient(bornDate:DATE,weight:W), @age(date,AGE), T:therapy(dru:DRUG,dose:DOSE),
  recommendedDose(drug:DRUG, dose:RD, minAge:MA, MinWeight:MW), AGE<MA, W<MW, DOSE>RD,
  MSG:="Prescribed dose " + DOSE + "exceed recommend dose " + RD, @newID(ID), @now(TIME).
```

The generated prescription error event must be properly handled in the process, for example an error message is visualized by means of a GUI to the physician.

```
ID:view_msg(method:"exception.jar", msg:MSG):-
   X:drugPrescription_medicalError(relativeTo:TASK, timestamp:TIME, msg:MSG), @newID(ID).
```

Queries can be also used for exploring clinical processes ontologies in a semantic fashion. For instance `malNeoplasm_f_patient(patient:P)?` returns every female patients suffering of any malignant neoplasm (e.g `P=#21, P=#34` ids are given for answer), where `malNeoplasm_f_patient(patient:P)`:

```
malNeoplasm_f_patient(patient:P):- P:patient(sex:#F),suffer(patient:P,disease:D),
D:malignant_neoplasm().
```

4 Implementation Issues

In this section the prototypical implementation of the proposed framework is presented. The prototype has been obtained by combining the JBPM engine [19] and the DLV+ system [20]. It is designed to follow a clinical processes life-cycle model based on 3 phases: *design*, *execution and monitoring*, *analytics*. Each phase is implemented by an ad-hoc software module. The *Design* module exploits the DLV+ system. It provides functionalities for importing and/or representing (by using direct "on-screen" drawing and manual specification facilities): medical ontologies, clinical processes and guidelines. For each clinical process a set of risk and error rules can be described in terms of ontology constraints and/or reasoning tasks. Acquired and/or represented schemas and instances are stored in a knowledge base and can be queried by using querying and meta-querying capabilities of DLV+ system. The *Execution & Monitoring* module is mainly constituted by the JBPM engine that interact with DLV+ system. Process execution is performed in two ways: (i) by using a *workflow enactment* strategy. In this case, a process schema, designed and stored in DLV+, is imported in JBPM and automatically executed involving actors that can be humans or machines (e.g. legacy systems supplying results of medical examinations); (ii) by using a dynamic *workflow composition* strategy. In this case, nodes to execute are selected step by step by choosing the most appropriate one in a given moment. Nodes are chosen by using semantic querying capabilities of DLV+ system and executed by JBPM. Queries allows to specify patient clinical data and each significant information available in the particular moment of the execution (e.g. drug interaction, allergies, etc). The execution generates process instances that are stored in the DLV+ knowledge base. Ad hoc extensions permit the interaction between JBPM and DLV+. So reasoning, querying and meta-querying over schemas and available instances are possible. This way, process execution can be monitored by running risk and errors rules that equip clinical process schemas. Events generated by rules alert process actors that can react for preventing risks and errors. The *Analytics* module aims at allowing analysis of the clinical processes instances after their acquisition. The execution of clinical processes makes available process instances that are also ontology instances. This way a large amount of semantically enriched data becomes available for retrieval, querying and analysis. Analysis are performed by creating reports obtained by semantic query of acquired process instances.

5 Conclusion and Future Work

This paper presented an ontology-based clinical knowledge representation framework that constitutes an innovative approach for semantic business process management in the field of healthcare information systems. Main features of the presented semantic approach (founded on logic programming) is that it, conversely to already existing systems and approaches, enables to represent clinical process ontologies that can be equipped with error and risk management facilities. In particular, the proposed framework allows to jointly manage declarative and procedural aspects of medical knowledge and express reasoning tasks that exploit represented knowledge in order to prevent errors that can create risks for patients. The framework enables, also: (i) manual process execution in which each clinical activity to execute in a given moment is chosen by physician on the base of the current configuration of patient and disease parameters, and (ii) automatic execution by means of the enactment of an already designed process schema (e.g. guidelines execution). During process execution, process and ontology instances are acquired and stored in a knowledge base. The execution can be monitored by running (over clinical process schemas and instances) reasoning tasks that implements error and risk rules. The practical application of the framework may allow better health care decision making capabilities that enables health care professionals to improve risks and errors prevention. But, currently the framework has been implemented in a prototypical way, so a wide practical and empirical evaluation is not yet available. The framework could be used in other application fields by changing adopted ontologies. Knowledge representation, however, is mainly made by hand so the effectiveness of the framework strongly depends from usability of adopted human-computer interfaces. The main challenging future research and development problems to face are: (i) the improvement of the query and enactment engine that works on the conjunct representation of workflows and ontologies; (ii) the definition of further monitoring and analytical technics based on workflow mining methods [25,26] that allows physicians to discover error and risk causes hidden in performed processes.

References

1. Classification Commune des Actes Médicaux (CCAM), http://www.ccam.sante.fr/
2. Current Procedural Terminology (CPT): a registered trademark of the American Medical Association (AMA), http://www.ama-assn.org/ama/pub/category/3113.html
3. International Classification of Diseases, Tenth Revision, Clinical Modification (ICD-10-CM), http://www.cdc.gov/nchs/about/otheract/icd9/icd10cm.htm
4. World Health Organization (WHO), http://www.who.int/classifications/en/
5. Logical Observation Identifiers Names and Codes (LOINC), http://loinc.org/
6. The Office of Population Censuses and Surveys Classification of Surgical Operations and Procedures, 4th Revision (OPCS-4), http://www.openclinical.org/medTermOPCS4.html
7. Systematized Nomenclature of Medicine-Clinical Terms, http://www.ihtsdo.org/snomed-ct
8. Medical Subject headings (MeSH), http://www.nlm.nih.gov/mesh/2008/index.html
9. Unified medical Language System (UMLS), http://www.nlm.nih.gov/research/umls/
10. Sackett, D.L., Rosenberg, W.M.C., Gray, M.J.A., Haynes, B.R., Richardson, S.W.: Evidence based medicine: what it is and what it isn't. BMJ 312(7023), 71–72 (1996)
11. Guide Line Interchange Format (GLIF), http://www.glif.org
12. Sutton, D.R., Fox, J.: The syntax and semantics of the proforma guideline modeling language. JAMIA 10(5), 433–443 (2003)

13. Pryor, T.A., Hripcsak, G.: The arden syntax for medical logic modules. Journal of Clinical Monitoring and Computing 10(4), 215–224 (1993)
14. Health Level Seven (HL7), http://www.hl7.org/
15. Musen, M.A., Tu, S.W., Das, A.K., Shahar, Y.: Eon: A component-based approach to automation of protocol-directed therapy. JAMIA 3(6), 367–388 (1996)
16. Shahar, Y., Young, O., Shalom, E., Mayaffit, A., Moskovitch, R., Hessing, A., Galperin, M.: Degel: A hybrid, multiple-ontology framework for specification and retrieval of clinical guidelines. In: Dojat, M., Keravnou, E.T., Barahona, P. (eds.) AIME 2003. LNCS, vol. 2780, pp. 122–131. Springer, Heidelberg (2003)
17. Kawamoto, K., Lobach, D.F.: Design, implementation, use, and preliminary evaluation of sebastian, a standards-based web service for clinical decision support. In: AMIA Symp., pp. 380–384 (2005)
18. Georgakopoulos, D., Hornick, M., Sheth, A.: An overview of workflow management: from process modeling to workflow automation infrastructure. Distrib. Parallel Databases 3(2), 119–153 (1995)
19. jBPM Process Definition Language (jPDL), http://www.jboss.org/jbossjbpm/jpdl/
20. Ricca, F., Leone, N.: Disjunctive logic programming with types and objects: The dlv+ system. J. Applied Logic 5(3), 545–573 (2007)
21. Eiter, T., Gottlob, G., Mannila, H.: Disjunctive Datalog. ACM Transactions on Database Systems 22(3), 364–418 (1997)
22. Leone, N., Pfeifer, G., Faber, W., Eiter, T., Gottlob, G., Perri, S., Scarcello, F.: The dlv system for knowledge representation and reasoning. ACM Trans. Comp. Logic 7(3), 499–562 (2006)
23. International Statistical Classification of Diseases and Related Health Problems 10th Revision, http://www.who.int/classifications/apps/icd/icd10online/
24. Anatomical Therapeutic Chemical (ATC) and Defined Daily Dose (DDD) system, http://www.whocc.no/atcddd/
25. Van der Aalst, W.M.P., Van Dongen, B.F., Herbst, J., Maruster, L., Schimm, G., Weijters, A.J.M.M.: Workflow mining: A survey of issues and approaches. Data Knowl. Eng. 47(2), 237–267 (2003)
26. Gaaloul, W., Alaoui, S., Bana, K., Godart, C.: Mining workflow patterns through event-data analysis. Saint-w, 226–229 (2005)

Ontology-Based Tools for Automating Integration and Validation of Firewall Rules

Ana-Maria Ghiran, Gheorghe Cosmin Silaghi, and Nicolae Tomai

Babeş-Bolyai University, Dept. of Business Information Systems
Str. Theodor Mihali 58-60, 400591, Cluj-Napoca, Romania
{Anamaria.Ghiran,Gheorghe.Silaghi,Nicolae.Tomai}@econ.ubbcluj.ro

Abstract. Firewalls are recognized as efficient instruments in deploying security in computer networks. But, they may become useless in cases when network administrators do not possess enough skills and expertise to properly configure them. Nowadays, firewall rules are integrated in the broader scope of enterprise security management. Thus, deriving correct and consistent rules for firewalls is mandatory and they need to be assimilated in the global security policy of the enterprise. In this paper we present tools for managing firewalls using ontologies and semantic-rich languages. With our approach, network managers can develop new firewall rules, automatically verify and validate their correctness and consistency and integrate them with previous existing rules.

Keywords: Ontologies, firewall management, rule-based reasoning.

1 Introduction

Securing the access control of an Intranet is one of the main objectives of all the network administrators and firewalls are among the most used tools in the first line of defense. Configuring firewalls proved to be a difficult task even for experienced practitioners and the threats are becoming more and more complex. One of the biggest problems the network administrators are dealing with regards the wrong configuration of the firewalls [1]. Often they fail to specify all the required rules that would enforce a certain level of security. Configuration of firewalls should be pursued in an automated way, otherwise this becomes error-prone and a time consuming activity.

Automating firewall management is hard to achieve because it is not easy to represent the domain knowledge. *Ontology*-based knowledge modeling represents a good alternative because they enable the separation of the meaning from the processing, allowing large collaborative knowledge bases to emerge. Ontologies are employed as a modeling tool for many domains where the integration and the uniform representation of information derived from various sources could be a challenge. W3C recommends the Web Ontology Language (OWL) [2] for ontologies used to describe the web but this standard is largely adopted in other application domains as well.

Despite the vast usage of ontologies in different areas, there are not too many ontology-based approaches in network administration and firewall management.

W. Abramowicz (Ed.): BIS 2009, LNBIP 21, pp. 37–48, 2009.

A global framework for automated firewall management should enable reliable firewall rules generation in compliance with the global security policy of the organization and the verification of actual rules for triggering out errors or rules inconsistencies. This paper presents tools toward building such a global framework.

In this paper we model the firewall configuration expertise with ontologies, using OWL [2] for knowledge representation. Our formal representation will allow formally disambiguating and structuring the represented knowledge. Further, we employ the Semantic Web Rule Language (SWRL) [3] for policy validation i.e. detecting firewall conflicts. The power of our tools will reside in a dual usage of reasoners. Ontology reasoners will allow integration of new rules with existing ones, while SWRL knowledge reasoners will allow validation and conflict detection.

Our approach provides new insights over firewall configuration proposing a solution for (i) generating reliable and conflict-free firewall configurations and (ii) analyzing the existing firewall configuration policies. Our solution further enhance the integration of firewall policy with other security policies, but this is beyond the scope of the current paper.

The paper is organized as follows. Section 2 introduces the firewall management problem and its challenges and reviews the current approaches for dealing with firewall configuration. Section 3 describes our ontology-based tools, presenting the ontology modeling and the SWRL rules for validation and conflict detection. Section 4 concludes the paper.

2 Background

In this section we define the firewall management problem and shortly brief related approaches.

2.1 The Firewall Management Problem

Firewalls manage the traffic between the public network and the private network zones or/and between private zones in the local network. Their main function is to examine all the packets that enter or leave the private network and to decide whether to accept them or not. Firewalls also can further be used for other functions like providing proxy services or network address translation (NAT). A standard firewall configuration usually consists of at least one router with access control capabilities at the boundary of the organization's network (the packet filter) and optionally more powerful firewalls located behind the router [4].

At a higher level, a firewall policy describes the company' security guidelines and the network objects on which the rules are applied: which computers are user desktops, web servers, file servers and so on, in order to confirm that the right rule is used for each one. At a lower level, packet filter firewalls employ rules for classifying packets based on their header fields [5]. The most commonly used fields are protocol type, source ip address, source port, destination ip address, destination port. Beside the filtering fields, a rule contains an action to

Table 1. Packet filter firewall rules

Rule No.	Protocol	Source address	Destination Address	Source Port	Destination Port	Action
1	TCP	123.12.0.0/16	22.33.44.0/24	any	80	accept
2	TCP	123.12.1.0/24	22.33.0.0/16	any	80	deny
3	TCP	123.12.1.0/24	22.33.44.1	any	80	deny
4	TCP	123.12.1.1	22.33.44.1	any	80	deny
5	TCP	123.12.0.0/16	22.33.0.0/16	any	any	accept

determine what to do with the matching packet and an ordering number to set the succession of activation. Table 1 exemplifies some packet filter rules.

Usually, firewalls have a big number of rules; their order is important and understanding and writing correct rules requires a minimum domain expertise. Manually managing firewall policy configurations become impractical for firewalls having thousands of rules. Big number of rules could be a source of conflicts and redundancies [6,7].

In general, firewall decisions are activated in sequence. In the case a packet matches more rules, only the first rule in the sequence is applied, the remaining ones being ignored. All the rules that have never been activated due to previous decisions are considered to be *shadowed*. Other rules are simply *redundant*, the same filtering policy being expressed at least twice in the firewall configuration file.

Hamed and Al-Shaer [6] derive a good taxonomy regarding the conflicts that could appear between the network security policies. They identified the following *critical errors*:

- shadowing - the shadowed rules are never activated (e.g. rule 4 in table 1 is shadowed by rule 1) and their reordering or removal is mandatory
- redundancy - redundant rules (rules 3 and 4 in table 1) just increase the configuration file and consequently the search time and memory requirements for rule matching. Because for many routers performance is mandatory [7], redundant rules need to be removed from the rules list.

Beside these, other conflicts named *warning errors*, listed below, are not so severe but must be considered because they influence the firewall policy [8]:

- generalization - a rule is a generalization of another rule if it can match all the packets that match another specific rule that precedes it (e.g. rule 5 is a generalization for rule 2). It is often used to exclude a specific traffic from a filtering action;
- correlation - two rules are correlated if the first one matches some packets that match also the second rule and the second rule matches some packets that match the first rule and the action of the two rules is different (rules 1 and 2). In case the order of the rules is reversed, the same traffic that before was denied could get accepted or the traffic that before was accepted could now be denied.

System administrators must configure and manage the firewalls according to a security policy that meets the company's needs. What is acceptable for a company might be unacceptable for others. Often the firewalls are configured in isolation, purely based on the network administrator expertise. This is a major drawback as they have to cooperate to emerge the combination of the local policies of devices that supplies the final security policy. In large organizations, where we can find firewalls deployed over multiple network components, a packet can be examined by many sets of firewalls at different times due to dynamic routing. Because the majority of the firewall policies are locally defined, it is very difficult to reason whether all these sets of firewalls satisfy the end-to-end security policies of the enterprise [9].

In this paper we tackle the problem of automating firewall management. For the scope of this paper, we include the identification of firewall conflicts, either errors or warnings and automating firewall rules generation and interoperability.

2.2 Related Work

Gouda and Liu [7] identified two approaches to reduce the firewall errors. The first one is to prevent the errors from the design time using high level languages. The second approach is to detect errors after firewalls have been designed and this can be done either using firewall queries [10] or employing different conflict detection methods. Among them, non-intrusive methods are preferred [11].

Regarding the first approach, we should note that firewall configuration files are often written in a vendor-specific language. The methods range from writing commands at a low level syntax to using a graphical management console. This is a problem when a new network administrator takes over or when there is a technical audit for the organization. Some approaches try to define some high level languages to specify the firewall rules, like the High Level Firewall Language (HLFL) [12]. Unfortunately, they still consider rules in sequence and do not prevent conflicts like shadowing and redundancy. DaxFi [13] is an alternative approach and allows writing rules in XML and then engineer or reverse-engineer. DaxFi supplies a way to remain independent from the syntax of a specific vendor and provides support for rules identification and comparison. But, it lacks on detection algorithms for the above-mentioned firewall conflicts.

In order to generate consistent firewall rules an idea is to generate them from the global security policies defined at a higher level in the organization [14]. Thus, common languages and knowledge representation models might be employed to enforce that every network administrator speaks the same language about the firewalls.

General policies are often described informally but policy languages exist for access control: XACML [15], X-RBAC [16], KAoS [17] or Ponder [18]. Both XACML and X-RBAC are based on XML, but lack on formal meaning support for the description of the entities or relationships. KAoS accomplished this to a certain level, by representing the policies using OWL. It provides policy and domain management services for agent and other distributed computing platforms. In KAoS, a policy constrains the actions that an agent is allowed or obliged to perform in a given context.

Another way to generate a complete set of rules is to use the decision diagrams employed for formally specifying the firewall [7]. Converting a decision diagram to a compact sequence of rules is automatically carried by a series of algorithms.

Regarding the second approach - analyzing firewalls after they have been designed, existing tools usually rely on algorithms for analyzing the firewall's access lists. Their objective is to remove the redundant rules, to notify the administrator about the existence of shadowed rules or about the exceptions generated by the rules' order.

Many works in this field are based on graph algorithms. Hazelhurst et al. [19] explored the use of binary decision diagrams (BDDs) for canonically representation of the access rule lists. Once the BDDs are constructed, verifying rules' equivalence can be done with little resources. The BDDs have the potential to speed up the search during the rule matching and to assist with the rules analysis. BDD have also been used by [9]. They designed a tool using static analysis for discovering the inconsistencies.

Al-Shaer and Hamed [8] proposed to use policy trees to visualize and find rule-set anomalies. A tree represent a rule. To discover anomalies in a rule-set it is enough to determine if any two rules coincide in the same policy tree path.

Moving toward a formal knowledge representation approach, Eronen and Zitting [11] use constraint programming to analyze the firewall's access lists. Besides the algorithms, they include a knowledge base about different network protocols, common configuration mistakes, some information about the network topology and also the access lists. Although this approach permits expanding the knowledge base, it still can not be integrated with other sorts of security policies because all system policies should be represented in the same specific way.

Fitzgerald [20] proposed a similar approach to incorporate the information needed for firewall decisions in a model. They use Description Logic to capture the domain knowledge and SWRL for reasoning. But their solution is tightened to the particular case of the Linux Netfilter firewall.

3 A Generic Framework for Firewall Management

In this section we define our ontology-based generic framework for firewall management.

3.1 Representing the Firewall Knowledge

To enhance an automatic method for firewall configuration design and validation we employ a formal description. This description provides us with a set of constructs to build a meaningful level of knowledge. We wrote our ontology in OWL with the support of the Protégé[1] tool [21].

Figure 1 depicts the firewall domain classes taxonomy. Classes depicted with solid color are complete classes. Given some instances, the ontology reasoner can infer whether these instances belong to the complete classes or not. Thus, the

[1] http://protege.stanford.edu

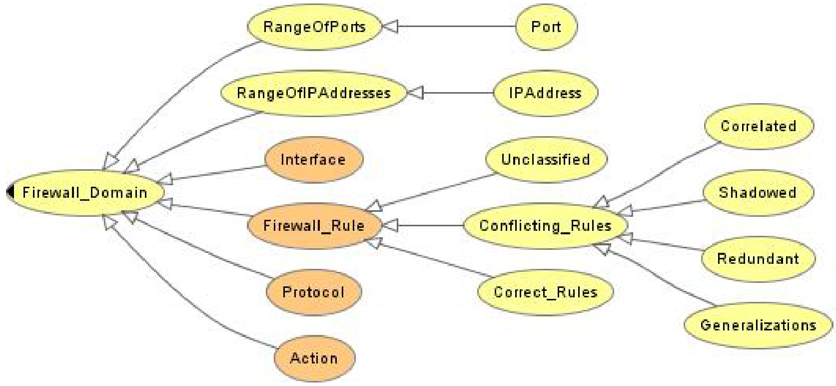


Fig. 1. The firewall domain classes taxonomy

ontology reasoner will be able to automatically identify a firewall rule, here of interest.

Central to our ontology, the `Firewall_Rule` represents the rules of the firewall. Figure 2 depicts all the logical conditions that fully describe the `Firewall_rule` class. These conditions are based on classes taken from the firewall domain (see figure 1). A rule is identified by a given number (the internal property `hasNo`). This identification number is useful for conflict analysis and for integrating new rules with existing rules. `Interface` models the incoming or the outgoing communication and it is defined through enumeration of its instances: `In` and `Out`, while `Action` models possible actions like `Accept`, `Deny` or `Permit` taken when the rule is matched. Filtered packets are communicated on a given protocol described by the `Protocol` concept. Each rule matches on source and destination `IP Addresses` and `Ports`.

Integrated in the internal security policy of the organization, firewall rules refer to `Network_Elements` (figure 3). Network elements are identified by their `IP Address`. A `Network_Element` might be an external (possibly malicious, un-

Fig. 2. The `Firewall_rule` class

known or collaborative) host, might refer to internal network elements or demilitarized zones (DMZs). Internal network elements, referred as `Internal_Network`, can be categorized in various security zones (in the picture we described only two security zones) with different access allowances.

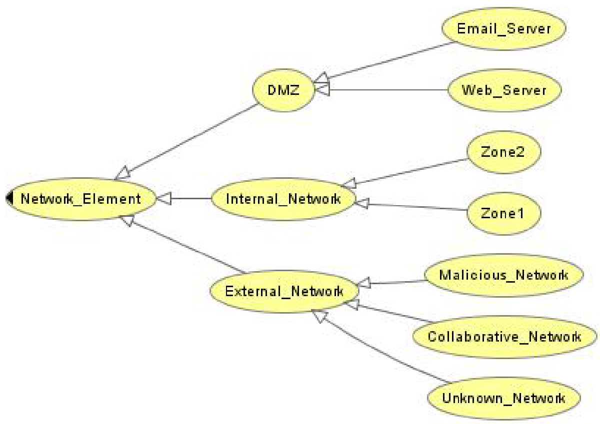


Fig. 3. The network elements taxonomy

Network elements concepts are related with the firewall rules through properties (see figure 4). Every network element has the `hasIPAddress` property, which was declared functional, meaning that the values are unique. Its inverse property is `isIPAddressFor`. Thus, the ontology reasoner can infer which network elements are subject to a firewall rule and for each network element the proper level of the security can be verified.

Fig. 4. The logical description of a network element

OWL was selected in order to enhance our model with automated reasoning facilities. Reasoning will permit us to derive new knowledge based on an initial set of rules. OWL, being described with XML, enhance direct integration with firewall configuration files written in XML syntax like DaxFi. Therefore, when putting the system live, the network manager has to import the existing firewall rules in the ontology editor and fill or write new rules.

Automated reasoning capabilities (like Pellet[2] [22], Racer[3] [23]) of the ontology editor allow ontology verification for inconsistencies. OWL allows us to define constraints. When such a constraint is violated by some data sets, the reasoner will report it as an inconsistency. Further, we can apply classification to decide whether a a class is a subclass of another class or if a property is a sub-property of another, to respond question like "Can a Network Identifier be an IP Address?". Also, we can infer if an individual is an instance of a class expression, responding questions like "Is an IP address recognized as a malicious one?"

Besides the firewall configuration information, our ontology includes data regarding the network topology and network objects, the description of the protocols and services and further induction can be performed with this respect.

3.2 Tools for Advanced Reasoning

A pure ontology based approach has difficulties with regard to what could be said about properties or the definition of variables. We can not define concepts that are assigned a value only at deployment or at run time (e.g. the values for IP addresses for instance). To solve this particular problem, we use SWRL rules [3]. SWRL extends the OWL axioms with Horn-like rules that can be combined with an OWL knowledge base. The SWRL rules consist of an implication between an antecedent (body) and a consequent (head), both of which are positive conjunctions of atoms. The intended meaning of a SWRL rule can be read as: whether the conditions specified in the antecedent hold, then the conditions specified in the consequent must also hold [3]. The atoms that compose the conjunctions can be of the form $C(x)$, $P(x,y)$, $sameAs(x,y)$ or $differentFrom(x,y)$, where C is an OWL description, P is an OWL property, and x, y are either variables, OWL individuals or OWL data values.

SWRL also provides so-called "built-ins" that allow user-defined methods to be used in rules. In the rules below we use some core built-ins - $swrlb : startsWith$ which returns true if the first argument starts with the second argument or $swrlb : greaterThanOrEqual$ which compares two values.

That means that new information can be generated related to the OWL individuals. For instance we can infer that "if there are two `IP Addresses` that have the same `Network Identifier` then they are in the same area". The inference rule for this statement is given in eq. 1:

$$\begin{aligned} IPAddress(?IP1) \wedge NetworkID(?Nx) \wedge \\ swrlb : startsWith(?IP1, ?Nx) \wedge \\ IPAddress(?IP2) \wedge NetworkID(?Ny) \wedge \\ swrlb : startsWith(?IP2, ?Ny) \wedge \\ sameAs(?Nx, ?Ny) \rightarrow sameArea(?IP1, ?IP2) \end{aligned} \tag{1}$$

We can derive rules that identify the conflicts in existing implemented policies. Eq. 3 and 4 exemplify the SWRL rules for the critical errors: shadowing and

[2] http://clarkparsia.com/pellet

[3] http://www.racer-systems.com/

redundancy rules. For testing the shadowing error (eq. 3), we identify two different rules $?x$ and $?y$ and we test for every possible condition regarding shadowing.

$$\begin{aligned} Rule(?x) \wedge hasAction(?ax, ?x) \wedge hasNo(?nrx, ?x) \wedge \\ hasInterface(?Ix, ?x) \wedge hasProtocol(?px, ?x) \wedge \\ hasSourceIPAddress(?IPsx, ?x) \wedge hasSourcePort(?ptsx, ?x) \wedge \\ hasDestIPAddress(?IPdx, ?x) \wedge hasDestPort(?ptdx, ?x) \\ \rightarrow IdentifiedRule(?x, ?nrx, ?ax, ?px, ?Ix, ?IPsx, ?ptsx, ?IPdx, ?ptdx) \end{aligned} \tag{2}$$

$$\begin{aligned} IdentifiedRule(?x, ?nrx, ?ax, ?px, ?Ix, ?IPsx, ?portsx, ?IPdx, ?ptdx) \wedge \\ IdentifiedRule(?y, ?nry, ?ay, ?py, ?Iy, ?IPsy, ?portsy, ?IPdy, ?ptdy) \wedge \\ swrlb : greaterThan(?nrx, ?nry) \wedge differentFrom(?ax, ?ay) \wedge \\ sameAs(?px, ?py) \wedge sameAs(?Ix, ?Iy) \wedge \\ include(?IPsx, ?IPsy) \wedge swrlb : greaterThanOrEqual(?ptsx, ?ptsy) \wedge \\ include(?IPdx, ?IPdy) \wedge swrlb : greaterThanOrEqual(?ptdx, ?ptdy) \\ \rightarrow shadows(?x, ?y) \wedge shadowed(?y) \end{aligned} \tag{3}$$

The redundancy SWRL rule described by eq. 4 identifies one original rule and all rules that are replicating the original rules.

$$\begin{aligned} IdentifiedRule(?m, ?nrm, ?am, ?pm, ?Im, ?IPsm, ?ptsm) \wedge \\ IdentifiedRule(?n, ?nrn, ?an, ?pn, ?In, ?IPsn, ?ptsn) \wedge \\ swrlb : greaterThan(?nrm, ?nrn) \wedge sameAs(?am, ?an) \wedge \\ sameAs(?pm, ?pn) \wedge sameAs(?Im, ?In) \wedge \\ include(?IPsm, ?IPsn) \wedge sameAs(?portsm, ?portsn) \wedge \\ \rightarrow original(?m) \wedge redundant(?n) \end{aligned} \tag{4}$$

SWRL rules for warning errors can be generated in a similar way. We omit them due to space restrictions.

SWRL rules together with the ontology model are loaded in a reasoner engine like JESS[4] [24], that can bring up all new or existing conflicting firewall rules. Taking as input a list of firewall configuration statements, JESS can match the logical inferences rules on them, triggering out the above-specified conflicts. For example, when we will try to add a new decision rule in the firewall configuration that refer to a specific IP Addresses for which we already had applied some decision rules, the framework automatically detects and classifies it as shadowed (if it has different actions) or redundant (if the actions are the same) or as an original good rule.

3.3 Automatic Maintenance of the Domain Model

Algorithm 1 presents the logical steps that should be performed toward an automated management of firewalls, by maintaining the firewall ontology in a consistent state.

[4] http://www.jessrules.com

Algorithm 1. Steps to be performed for automated firewall management

```
the network administrator define the organization needs
firewall ontology is generated
repeat
    ontology reasoners check for inconsistencies and infer new instances or classes
    correct the ontology's inconsistencies and assert some inferred classes or individuals
    apply SWRL-based reasoning to detect the conflicts
    incorporate new knowledge in the ontology
until the system ends its life
```

The first step requires the network administrator to state the needs of his organization: the services enabled for external users (the range of open ports), the network topology (elements), which are the communication destinations of the internal users (the range of IP Addresses) and by what means the internal users perform the communication (e.g. web browsing). Next, the firewall ontology is generated. After that, the ontology is maintained in a consistent state with the support of the ontology and knowledge reasoners. Conflicting rules are identified and removed from the ontology and new inferred knowledge is transfered back into the ontology. The ontology evolves during time, as the communication needs of the organization shift.

The steps presented in algorithm 1 assume that the network administrator builds an ontology from scratch. But, the steps depicted in the `repeat-until` loop can be applied if one wants to integrate new rules within an existing firewall or when several firewalls should be integrated.

The performance on applying the ontology and SWRL reasoning on large rule sets strictly depends on the performance of the reasoners. Selecting powerful reasoning engines might help in getting a good performance, but is out of the scope of this paper to compare various reasoners. More, designing firewalls is a activity that happens off-line, only from time to time. Thus even with a slow reasoner, the network administrator will gets useful results.

4 Conclusion

In this paper, we proposed an ontology-based model to represent the firewall configuration knowledge. Our approach relies on devising an ontology for the firewall domain and equipping it with logical inference rules. This facilitates the reasoning process over the structure and relationships between various decision policies implemented at firewall level. It also enables the classification and comparison between firewall rules. The existence of such a model is an important feature in medium and large organizations where there are tens or hundreds of different firewall rules sets, designed in a bottom-up fashion by various network administrators.

We presented tools based on two sorts of reasoning: ontology reasoning and rule based reasoning. We got this separation, because nowadays, most ontology reasoners do not support SWRL rules. Reasoning integration is obtained using a knowledge reasoning engine. The knowledge reasoning engine allows further knowledge to be inferred and the automatically expansion of our ontology.

OWL features enhance the interoperability among different policies. In the same time, a rule-based language like SWRL provides us with the ability to validate the firewall policies and complements the OWL language in this way.

Acknowledgments. This work is supported by the Romanian Authority for Scientific Research under project IDEI_573.

References

1. Wool, A.: A quantitative study of firewall configuration errors. Computer 37(6), 62–67 (2004)
2. W3C Consortium: Web ontology language (consulted on 28 January 2009), http://www.w3.org/2004/OWL/
3. Horrocks, I., Patel-Schneider, P.F., Boley, H., Tabet, S., Grosof, B., Dean, M.: SWRL: A Semantic Web Rule Language Combining OWL and RuleML. W3C Member Submission (2004)
4. Wack, J., Cutler, K., Pole, J.: Guidelines on Firewalls and Firewall Policy: Recommendations of the National Institute of Standards and Technology. NIST special publication 800-41 (2002)
5. Adiseshu, H., Suri, S., Parulkar, G.M.: Detecting and resolving packet filter conflicts. In: IEEE INFOCOM 2000, The Conference on Computer Communications, vol. 3, pp. 1203–1212. IEEE Computer Society, Los Alamitos (2000)
6. Hamed, H., Al-Shaer, E.: Taxonomy of conflicts in network security policies. IEEE Communications Magazine 44(3), 134–141 (2006)
7. Gouda, M.G., Liu, A.X.: Structured firewall design. Computer Networks 51(4), 1106–1120 (2007)
8. Al-Shaer, E., Hamed, H.: Management and translation of filtering security policies. In: IEEE Intl. Conf. on Communications, pp. 256–260. IEEE Computer Society, Los Alamitos (2003)
9. Yuan, L., Mai, J., Su, Z., Chen, H., Chuah, C.N., Mohapatra, P.: Fireman: A toolkit for firewall modeling and analysis. In: IEEE Symposium on Security and Privacy, pp. 199–213. IEEE Computer Society, Los Alamitos (2006)
10. Liu, A.X., Gouda, M.G., Ma, H.H., Ngu, A.H.: Firewall queries. In: Higashino, T. (ed.) OPODIS 2004. LNCS, vol. 3544, pp. 197–212. Springer, Heidelberg (2005)
11. Eronen, P., Zitting, J.: An expert system for analyzing firewall rules. In: 6th Nordic Workshop on Secure IT Systems, Technical report IMM-TR-2001-14. Technical University of Denmark (2001)
12. The HLFL project (consulted on 28 January 2009), http://www.hlfl.org/
13. The Dynamic XML firewall (consulted on 28 January 2009), http://daxfi.sourceforge.net
14. Guttman, J.D.: Filtering postures: local enforcement for global policies. In: IEEE Symposium on Security and Privacy, pp. 120–129. IEEE Computer Society, Los Alamitos (1997)
15. Moses, T.: eXtensible Access Control Markup Language, Ver. 2.0. OASIS Standard (2005)

16. Joshi, J.B.D., Bhatti, R., Bertino, E., Ghafoor, A.: Access-control language for multidomain environments. IEEE Internet Computing 8(6), 40–50 (2004)
17. Uszok, A., Bradshaw, J.M., Johnson, M., Jeffers, R., Tate, A., Dalton, J., Aitken, S.: KAoS policy management for semantic Web Services. IEEE Intelligent Systems 19(4), 32–41 (2004)
18. Damianou, N., Dulay, N., Lupu, E., Sloman, M.: The PONDER policy specification language. In: Sloman, M., Lobo, J., Lupu, E.C. (eds.) POLICY 2001. LNCS, vol. 1995, pp. 18–38. Springer, Heidelberg (2001)
19. Hazelhurst, S., Fatti, A., Henwood, A.: Binary decision diagram representations of firewall and router access lists. Technical report, Department of Computer Science, University of the Witwatersrand, Johannesburg, South Africa (1998)
20. Fitzgerald, W.M., Foley, S.N., Foghlú, M.Ó.: Network access control interoperation using Semantic Web techniques. In: WOSIS 2008: 6th Intl. Workshop on Security in Information Systems, pp. 26–37. INSTICC Press (2008)
21. Knublauch, H., Fergerson, R.W., Noy, N.F., Musen, M.A.: The Protégé OWL plugin: An open development environment for Semantic Web applications. In: McIlraith, S.A., Plexousakis, D., van Harmelen, F. (eds.) ISWC 2004. LNCS, vol. 3298, pp. 229–243. Springer, Heidelberg (2004)
22. Sirin, E., Parsia, B., Grau, B.C., Kalyanpur, A., Katz, Y.: Pellet: A practical OWL-DL reasoner. Technical report, University of Maryland (2005)
23. Haarslev, V., Möller, R.: Racer: An OWL reasoning agent for the Semantic Web. In: Intl. Workshop on Applications, Products and Services of Web-based Support Systems, in conjunction with the 2003 IEEE/WIC Intl. Conf. on Web Intelligence, pp. 91–95 (2003)
24. Frieman-Hill, E.: Jess in Action: Java Rule-Based Systems. Manning Publications (2003)

Ontological Mapping of Information Security Best-Practice Guidelines

Stefan Fenz, Thomas Pruckner, and Arman Manutscheri

Vienna University of Technology and Secure Business Austria, Vienna, Austria
fenz@ifs.tuwien.ac.at

Abstract. Due to a rapid growth in the use of electronic data processing and networking, an information security management system with a holistic and widespread view becomes more and more important for any kind of organization. The fundamental challenge for such systems is the representation and management of information security knowledge. While information security ontologies already exist, no methods have been proposed to map existing best-practice guidelines or information security standards to an existing ontology. Therefore, this paper presents a method for mapping the information security knowledge of the French EBIOS standard and the German IT Grundschutz Manual to a OWL-DL security ontology. Applying the introduced method allows to reuse existing information security knowledge bases and to map them to open and standardized data structures which can be easily reused by organizations and developers to support their existing information security management systems.

Keywords: Ontological mapping, information security best-practice guidelines, security ontology, EBIOS, IT Grundschutz Manual.

1 Introduction

In recent years a rapid growth in the use of electronic data processing and networking took place. By now almost all kind of organizations are depending on IT systems in large parts of their business activity. With the extensive use of information technologies and the increasing networking in all business areas the requirements on IT security widened dramatically [11,1]. The large quantity of potential threats and the growing complexity of IT systems led to the conclusion that the holistic perspective of IT security and the implementation of IT security management systems is absolutely essential [10]. The fundamental challenge for such systems is the representation and management of information security knowledge. The characteristics of an ontology allow to address this challenge. While ontologies in the information security domain exist (cf. [6,7,12]) no methodology has been proposed to map information security knowledge from existing information security standards or best-practice guidelines to these knowledge models. In this paper we propose a methodology for mapping information security best-practice guidelines to existing information security ontologies. Applying the introduced method allows to reuse existing information

W. Abramowicz (Ed.): BIS 2009, LNBIP 21, pp. 49–60, 2009.

security knowledge bases and to map them to open and standardized data structures which can be easily reused by organizations and developers to support their existing information security management systems. The proposed methodology is demonstrated by mapping the French EBIOS [5] and the German IT Grundschutz Manual [4] to the security ontology by [6].

2 Ontological Mapping of Information Security Best-Practice Guidelines

An essential requirement for mapping existing information security best-practice guidelines to an ontological structure is that the selected best-practice guideline is available in a machine-readable form. A survey among existing information security standards and best-practice guidelines has shown that national guidelines such as the German IT Grundschutz Manual [4] and the French EBIOS [5] are available in a machine-readable form. While EBIOS provides its knowledge base in form of structured XML-documents, the IT Grundschutz Manual provides a proprietary but still readable database structure. We propose the following methodology to map machine-readable information security best-practice guidelines to existing ontological structures:

- **Ontology analysis:** Before starting the actual mapping process, the ontological structure of the selected security ontology has to be analyzed. Especially the analysis of existing concepts and corresponding relations is crucial for relating them to the knowledge base structure identified in the next phase.
- **Knowledge base analysis:** This phase identifies entities and relations which are semantically similar to the ontological concepts and relations identified in the previous phase.
- **Mapping concepts and relations:** Based on the results of the previous two phases, this phase maps entities and relations of the machine-readable best-practice guideline representation to the ontological model.
- **Mapping the knowledge:** The mapping schema of the previous phase is used to map the actual knowledge from the best-practice guideline to the ontological information security model.
- **Evaluation:** Since the mapping of the knowledge may be conducted semi - automatically, the evaluation phase requires the manual evaluation of the mapped knowledge by human beings.

In the following sections we describe the application of the proposed mapping methodology and the difficulties which arise in the mapping process. We used the machine-readable knowledge bases of EBIOS and the IT Grundschutz Manual to map them to the security ontology by [6].

3 Ontology Analysis – Security Ontology

Figure 1 shows the high-level concepts (boxes) and corresponding relations (arrows represent at their start the domain and at their end the range of the corresponding relation) of the used security ontology (cf. [6] for further details on the used security ontology). A threat gives rise to follow-up threats, represents a potential danger to the organization's assets and affects specific security attributes (e.g. confidentiality, integrity, and/or availability) as soon as it exploits a vulnerability in the form of a physical, technical, or administrative weakness. Additionally each threat is described by potential threat origins (human or natural origin) and threat sources (accidental or deliberate source). For each vulnerability a severity value and the asset on which the vulnerability could be exploited is assigned. Controls have to be implemented to mitigate an identified vulnerability and to protect the respective assets by preventive, deterrent, recovery, or detective measures (control type). Each control is implemented as asset concept, or as combinations thereof. The controls are modeled on a highly granular level and are thus reusable for different standards. When implementing the controls, a compliance with various information security standards is implicit. The coded ontology follows the OWL-DL (W3C Web Ontology Language) [14] standard and ensures that the knowledge is represented in a standardized and formal form.

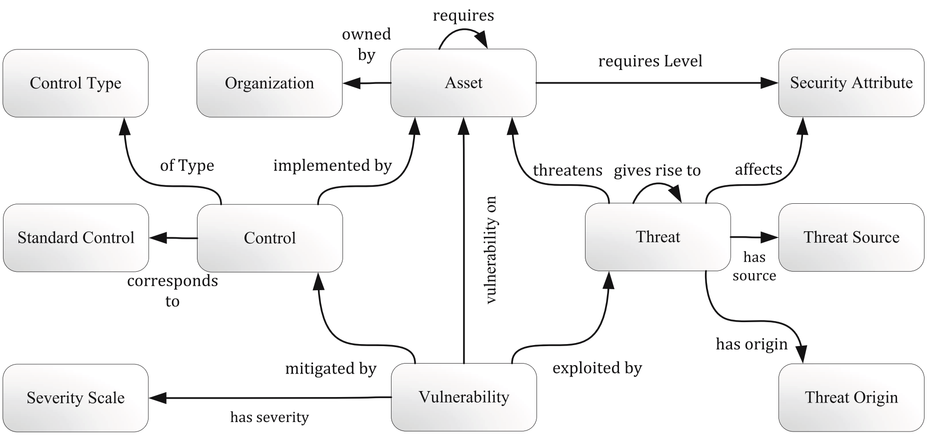


Fig. 1. Security ontology top-level concepts and relationships

4 Mapping EBIOS to the Security Ontology

According to the proposed methodology, this section shows how we mapped the knowledge represented by EBIOS to the security ontology.

4.1 Knowledge Base Analysis

EBIOS [5] was created by the DCSSI (Direction Centrale de la Sécurité des Systèmes d'Information - a department of the French Ministry of Defense) and represents a method for the assessment and treatment of IT security risks. For the definition of a certain level of security, EBIOS specifies generic security objectives that are used for the protection of entity types (assets) and the mitigation of vulnerabilities. The implementation of these objectives is carried out by predefined functional security requirements derived from standards like ISO 17799. The data-sets offered by the EBIOS method include descriptions of entity types, threats, vulnerabilities, and security objectives which can be achieved by the implementation of corresponding measures. See the EBIOS documentation [5] for further details.

4.2 Mapping Concepts and Relations

Figure 2 gives an idea of the relations between EBIOS and the security ontology but it is insufficient in order to map the provided information exactly. Therefore, Table 1 lists all mappable XML-elements and attributes defined by EBIOS and quotes their corresponding OWL-concepts and relations in the security ontology. The creation of such a table requires the semantic analysis of concepts and relations located in the source (best-practice guideline) and the target (security ontology). Although dictionary-based approaches can be used to map common keywords, this phase has to be conducted mainly by manual means. Especially the analysis of the concepts' and relations' natural language descriptions is important for an appropriate mapping between source and target.

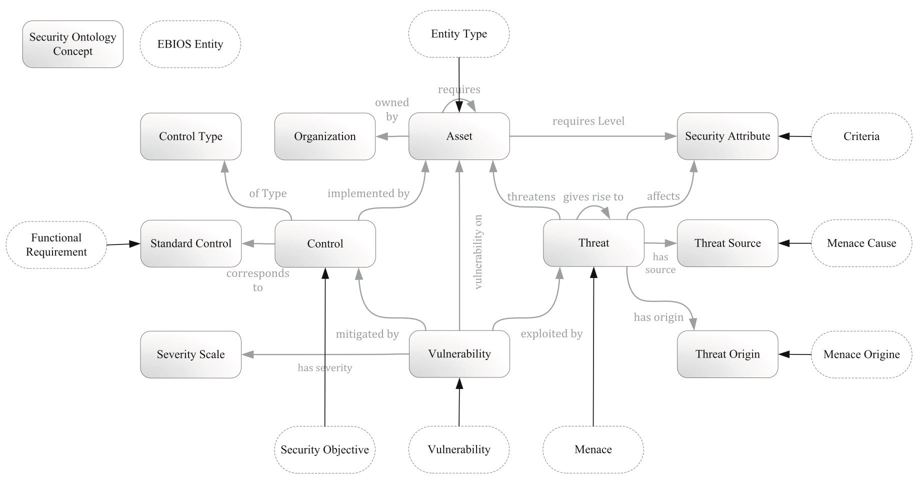


Fig. 2. Relationships among EBIOS entities and security ontology concepts

4.3 Mapping the Knowledge

In this section we show, by using the example of the fire threat, how we have mapped information security knowledge from EBIOS to the security ontology. As listed in Table 1, a menace in EBIOS is equivalent to a threat in the security ontology. The following code snippet shows the EBIOS XML representation of the fire threat[1].

```
<Menace ID="Menace.1050382052535" label="01-FIRE" selected="" description="Type:
     Natural/Human/Environmental  Accidental cause: Concentration of flammable or
     explosive..." justification="" descriptionMenaceElement="" potentiel="">
  <MenaceThemeList ID="MenaceThemeList.1050973114465">
    <Theme id="Theme.1013467459833" comments=""/>
  </MenaceThemeList>
```

[1] To enhance the readability we shortened the description text.

Table 1. EBIOS entities and attributes and their corresponding security ontology concepts and relations

EBIOS XML-elements and attributes	**Security ontology concepts and relations**
<EntityType> type description	ent:Asset subclasses of ent:Asset ent:description of abstract instances of subclasses of ent:Asset
<Vulnerability> label menace	sec:Vulnerability subclasses of sec:Vulnerability sec:exploitedBy of sec:Vulnerability
<EntityTypeList>	sec:threatens of sec:Threat
<Menace> label description	sec:Threat subclasses of sec:Threat sec:description of abstract instances of subclasses of sec:Threat
<SeverityScale>	sec:affects of sec:Threat
<Criteria> label description	sec:SecurityAttribute instances of sec:SecurityAttribute sec:description of abstract instances of subclasses of sec:SecurityAttribute
<MenaceCauseList> <MenaceCause> label description	sec:hasSource of sec:Threat sec:ThreatSource subclasses of sec:ThreatSource sec:description of abstract instances of subclasses of sec:ThreatSource
<MenaceOrigineList> <MenaceOrigine> label description	sec:hasOrigin of sec:Threat sec:ThreatOrigin subclasses of sec:ThreatOrigin sec:description of abstract instances of subclasses of sec:ThreatOrigin
<SecurityObjective> label content <SecurityObjectiveCovers>	sec:Control subclasses of sec:Control sec:description of abstract instances of subclasses of sec:Control sec:mitigatedBy of sec:Vulnerability
<FunctionnalRequirement> abbreviation description <Objective>	iso:Control iso:controlTitle of abstract instances of iso:Control iso:controlDescription of abstract instances of iso:Control sec:correspondsTo of iso:Control

```
  <SeverityScale ID="SeverityScale.1050973114465">
    <MenaceSeverity ID="MenaceSeverity.1109436174044" criteria="Criteria
         .1013307741641" severity="" violation="true"/>
    <MenaceSeverity ID="MenaceSeverity.1109108597320" criteria="Criteria
         .1011680648037" severity="" violation="true"/>
  </SeverityScale>
  <MenaceCauseList ID="MenaceCauseList.1050973114465"/>
    <MenaceCause id="MenaceCause.1012606157332" comments=""/>
    <MenaceCause id="MenaceCause.1011656568285" comments=""/>
  </MenaceCauseList>
  <MenaceOrigineList ID="MenaceOrigineList.1050973114465"/>
    <MenaceOrigine id="MenaceOrigine.1051413282991" comments=""/>
    <MenaceOrigine id="MenaceOrigine.1052902060343" comments=""/>
    <MenaceOrigine id="MenaceOrigine.1050514650356" comments=""/>
  </MenaceOrigineList>
</Menace>
```

The menaces' attribute *Label* and *Description* correspond to the threat sub-concepts and their descriptions. The element *SeverityScale* lists all affected *Criteria* which comply with *sec:SecurityAttribute* in the security ontology. The elements *MenaceCauseList* and *MenaceOrigineList* provide information about the sources and the origin of a threat. In the given example the attribute *Label* of the fire menace corresponds to the sub-concept *sec:Fire* in the security ontology. The element *SeverityScale* corresponds to the relation *sec:affects*. It lists the affected *Criteria* which correspond to *sec:SecurityAttribute* in the security ontology. The affected criteria in this example are 'Criteria.1013307741641' which is defined as availability and 'Criteria.1011680648037' which is defined as integrity. The element *MenaceCauseList* lists possible causes of a fire and is equivalent to the relation *sec:hasSource*. It is listing all possible *MenaceCause* elements which comply to the sub-concepts of *sec:ThreatSource* in the security ontology. In this example 'MenaceCause.1012606157332' stands for an accidental threat source and 'MenaceCause.1011656568285' stands for a deliberate threat source of a fire. The element *MenaceOrigineList* is equivalent to the relation *sec:hasOrigin*. It is listing all possible *MenaceOrigine* elements which comply with the subclasses of *sec:ThreatOrigin* in the ontology. The menace origins listed are 'MenaceOrigine.1051413282991', 'MenaceOrigine.1052902060343' and 'MenaceOrigine.1050514650356' which stand for environmental, human and natural threat origins. In the ontology the natural and environmental origins are summarized under the term natural origin. On the brief example of the fire threat we showed how to map EBIOS elements such as entity types, vulnerabilities, security objectives, and criteria to the security ontology. The structured XML knowledge representation and the developed mapping table (see Table 1) allowed us the semi-automatic mapping of the knowledge.

5 Mapping the IT Grundschutz Manual to the Security Ontology

According to the proposed methodology, this section shows how we mapped the IT Grundschutz knowledge to the security ontology. In contrast to the EBIOS mapping, the IT Grundschutz mapping requires substantial manual intervention.

5.1 Knowledge Base Analysis

IT Grundschutz is a holistic concept, helping SMEs to create an IT security level that is adequate to satisfy average protection requirements. It has been developed and published by the German Federal Office for Information Security (BSI). The IT Grundschutz Manual contains 3 main catalogs: (i) the modules-catalogs describe the typical aspects and applications for IT security, (ii) the threat-catalogs consist of five sub-catalogs which present numerous threat scenarios, and (iii) the safeguard-catalogs provide detailed safeguard implementation guidelines.

5.2 Mapping Concepts and Relations

Figure 3 shows the relations between IT Grundschutz and the security ontology. Since IT Grundschutz provides its very broad knowledge in a very flat structure (only three catalogs are used), only four Grundschutz entities have been mapped to the security ontology concepts. While entities 'Safeguard' and 'Threat' are mapped directly to the security ontology concepts 'Control' and 'Threat', the entities 'Module' and 'ISO 27001' have been mapped indirectly via support documents (e.g. BSI cross-reference tables). For each module (Generic Aspects, Infrastructure, IT Systems, Networks, etc.) a cross-reference table exists. Each cross-reference table lists threats relevant to the module and shows which safeguards can be used to mitigate the given threat. This enables us to establish the required links between Asset (Module), Control (Safeguard), and Threat (Threat). Since the structure of the IT Grundschutz does not exactly fit the structure of the security ontology, numerous manual actions have to ensure that the knowledge is appropriately incorporated into the security ontology:

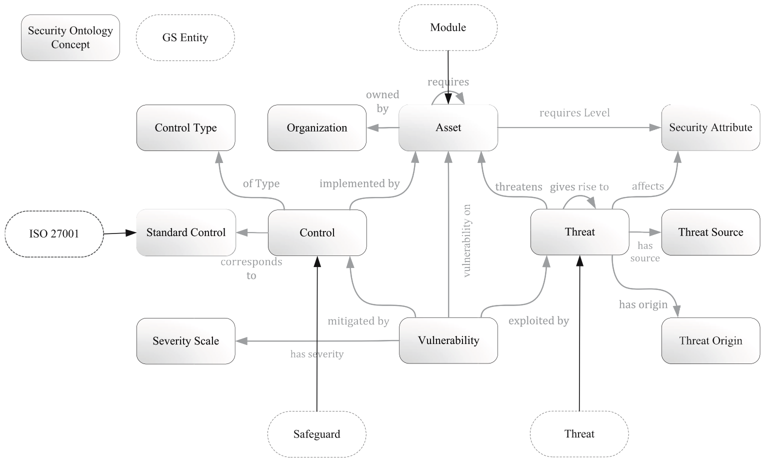


Fig. 3. Relationships among Grundschutz entities and security ontology concepts

1. The BSI cross-reference tables for each IT Grundschutz module are the starting point for the knowledge mapping process: threats and safeguards relevant to the considered module (asset) are identified.
2. IT Grundschutz safeguards are incorporated into the security ontology as controls and the natural language safeguard implementation description is manually transformed into a formal description (relation *sec:implementedBy*). Each control is connected by relation *sec:ofType* to an appropriate control type and the newly created control is related to an ontological representation of its original IT Grundschutz safeguard (relation *sec:correspondsTo*).
3. IT Grundschutz threats are incorporated into the security ontology. Related threats are connected by the *sec:giveRiseTo* relation. Threats directly threatening the considered asset are connected by the *sec:threatens* relation to the considered asset concept. Affected security attributes (e.g. confidentiality) are modeled by the *sec:affects* relation. Threat origin and source are modeled by the relations *sec:threatOrigin* and *sec:threatSource*. Since the IT Grundschutz does not provide structured information on these issues, we had to derive and incorporate them manually.
4. As IT Grundschutz does not provide knowledge on vulnerabilities, we had to derive them and their severity from the corresponding controls. The newly created vulnerability is connected by *sec:mitigatedBy* to corresponding controls and by *sec:exploitedBy* to corresponding threats. Depending on the considered module, relation *sec:vulnerabilityOn* connects the vulnerability to its 'sphere of action'.
5. At last, we have used the BSI ISO27001 - IT Grundschutz mapping tables to link the established controls to ISO 27001 controls.

5.3 Mapping the Knowledge

In this section we show, by using the example of the common server module (B 3.101), how we have mapped the information security knowledge from IT

Grundschutz to the security ontology. The subsequent steps correspond to the mapping process of the previous section.

1. Considered module: common server module (B 3.101) which is equivalent to the *ent:ComputerServer* security ontology concept (ent:ComputerServer $\sqsubseteq$ ent:Asset). Potential IT Grundschutz threat and safeguard: *Disruption of power supply (G 4.1)* and *Local uninterruptible power supply (M 1.28)*.
2. The IT Grundschutz safeguard *Local uninterruptible power supply (M 1.28)* has been mapped to the control *sec:UninterruptiblePowerSupplyControl*. The *sec:implementedBy* relation connects the control to the implementation concept *ent:UninterruptedPowerSupplyUnit* → the control is implemented if an uninterrupted power supply unit exists in the considered context. Furthermore, the control has been connected by *sec:controlType* to the *sec:PreventiveControlType* concept. Relation *sec:correspondsTo* connects it to the ontological representation of the original IT Grundschutz control (*gshb:M_1_28*).
3. Threat *Disruption of power supply (G 4.1)* has been mapped to the security ontology concept *sec:PowerLoss* (*sec:PowerLoss* $\sqsubseteq$ *sec:Threat*). The power loss threat is connected by the *sec:givesRiseTo* relation to the already existing *sec:ITComponentsDamage* threat. Based on the natural language threat description the power loss threat has been classified as a threat with a human or natural threat origin, and deliberate or accidental threat source respectively.
4. The vulnerability *sec:NoUninterruptiblePowerSupply* has been created. The *sec:exploitedBy* relation connects it to the *sec:PowerLoss* threat. The relation *sec:mitigatedBy* connects it to the *sec:UninterruptiblePowerSupplyControl* control. The relation *sec:vulnerabilityOn* restricts the vulnerability's sphere of action to the *ent:ComputerEquipmentAndAccessories* concept.
5. By using the BSI ISO 27001 - IT Grundschutz Mapping tables we were able to correspond the *sec:UninterruptiblePowerSupplyControl* control to the ISO 27001 controls *iso:A.9.2.1* and *iso:A.9.2.2*.

6 Difficulties in the Mapping Process

The following problems and incompatibilities had to be solved and compensated during the process of mapping EBIOS and the IT Grundschutz Manual to the security ontology:

Identification of Already Existing Concepts in the Ontology: The main problem when mapping several information security best-practice guidelines is the identification of already existing concepts in the ontology. One approach is to automatically search for terms existing in both, the considered best-practice guideline and the security ontology. However, the found items must be considered in detail if they really correspond to the respective counterpart. If the search results return no corresponding terms, existing concepts of the security ontology must be scanned for analogies by manual means.

No Concept for Vulnerabilities: The IT Grundschutz Manual does not work with the concept of vulnerabilities, unlike the NIST Handbook [8] on which the security ontology structure has been built. Therefore, vulnerabilities had to be created artificially from the scratch. Our approach is based on the NIST Handbook which states: *vulnerabilities are often analyzed in terms of missing safeguards*. Therefore, vulnerabilities were derived from the existing IT Grundschutz controls by implication. For example, interpreting the control *fire doors* as *fire doors should be in place*, the derived vulnerability would be *no fire doors*. This mapping mechanism enables the incorporation of the IT Grundschutz Manual knowledge in the security ontology while keeping its knowledge model consistent.

Vague Connections between Threats and Controls: The problem was to create clear relations between a threat and the corresponding control, which initially was not possible due to the structure of the IT Grundschutz Manual. As a solution 72 cross-reference tables, one for each IT Grundschutz Manual module, were used to identify the connections between threats and corresponding controls to get a more structured access to the relations.

No Relations between Threats: Unfortunately, EBIOS and the IT Grundschutz Manual do not describe connections between individual threats. Therefore, further information security standards, best-practice guidelines and expert knowledge had to be used to model them. To simplify this process a few top-level threats were identified (e.g. data disclosure, data tampering, and data loss) affecting certain security attributes (confidentiality, integrity, and availability).

Inconsistent Granularity of Information: Since the production of a consistent knowledge base with a similar grade of information detail is aimed for, the information of the IT Grundschutz Manual had to be filtered and changed, and topics covering very specific topics were left out in the mapping process. The mapping of topics mentioned in the BSI ISO 27001 - IT Grundschutz Mapping tables were defined as the minimum for the mapping process.

7 Evaluation

According to [13], informal and formal competency questions have been used to evaluate our ontology with the help of a team of experienced information security professionals. Since most ontology evaluation approaches, as described in [2], [9], or [3], are concerned with selecting the most appropriate ontology from a set of existing ontologies, the approach by [13] has been adopted to create an evaluation methodology which is able to check an ontology against its initial requirements. Therefore, the following evaluation phases have been conducted: (i) identification of informal competency questions based on best-practice guidelines and domain expert interviews, (ii) creation of formal competency questions based on the informal competency questions identified in the previous step, and (iii) evaluation (conducted by domain experts) of the formal competency question result sets. As domain experts are central to the ontology evaluation methodology, a team of eight information security professionals was put together. Although this is neither a significant nor representative group of experts, it helped improving the modeled information security knowledge. The following subsections show by an exemplary competency question how the evaluation has been conducted. For a full description of the evaluation process see [6].

7.1 Informal Competency Question

Since the security ontology has been designed to support the information security risk management process the domain expert team developed competency questions according to the generically defined information security risk management phases. The following exemplary competency question is used to show the conducted evaluation process:

Which vulnerabilities are exploited by a given threat and which controls can be used to mitigate the vulnerabilities?

7.2 Formal Competency Question

If a threat is threatening crucial assets of the considered organization, it has to be known which of the existing vulnerabilities the threat exploits and how these vulnerabilities can be mitigated by appropriate controls to reduce the risk to an acceptable level. First of all, the subsequent SPARQL statement queries the vulnerabilities which are associated by relation *sec:exploits* with the power loss threat. Note that the power loss threat is just an example and that the vulnerabilities of each threat can be revealed in the same way.

```
SELECT ?vulnerability
WHERE {sec:PowerLoss sec:exploits ?vulnerability}
```

Since one vulnerability of power loss is the unavailability of an uninterruptible power supply unit, the following query reveals the associated controls.

```
SELECT ?control
WHERE {sec:NoUninterruptiblePowerSupply sec:mitigatedBy ?control}
```

With the appropriate control concept on hand, the organization is now able to derive the control implementation descriptions to mitigate the corresponding vulnerability in the context of a given asset.

7.3 Result Set

By the implementation and the subsequent execution of the formal competency question set, each competency question resulted in a data set, which is evaluated by the security professional expert team in this evaluation step. The formal competency questions return formalized knowledge fragments (e.g., sec:UninterruptiblePowerSupplyUnit to mitigate the no uninterruptible power supply vulnerability and the corresponding power loss threat). Due to the high degree of complexity, not all formal competency questions have been answered with simple ontology queries. Nevertheless, it could be shown that the enriched ontology is able to answer such complex questions, even if an external calculation is required.

8 Conclusion

The more and more comprehensive use of electronic data processing and networking demands for giving particular attention to an IT security management solution capable of providing and dealing with information security knowledge regarding potential threats, vulnerabilities, and controls. We proposed a method for mapping information security best-practice guidelines to existing security ontologies. The method has been demonstrated by mapping EBIOS and the IT Grundschutz Manual to the security ontology: entities and their attributes defined in both knowledge bases have been assigned to corresponding concepts and relations defined in the security ontology. By means of this mapping schema the knowledge provided by EBIOS and the IT Grundschutz Manual can be transformed into OWL-code used by the security ontology. The introduced method for mapping information security knowledge is a guideline trying to equip existing security ontologies with widely accepted information security knowledge. The limitations of the developed method are: (i) in the case of unstructured knowledge sources (e.g. IT Grundschutz) it requires a lot of manual intervention and does not provide a satisfactory degree of automation, (ii) the attempt of incorporating more than one best-practice guideline has shown the limits of the methodology → even if one knowledge source can be semi-automatically incorporated it requires substantial manual intervention to map a further knowledge base on an existing body of knowledge. Further work focuses on addressing these issues and includes the mapping of further information security best-practice guidelines and standards to provide the community with a wide ontological information security knowledge base. Potential applications of such a knowledge base include but are not limited to risk management and automated compliance checks regarding information security standards such as ISO 27001.

Acknowledgments

This work was supported by grants of the Austrian Government's FIT-IT Research Initiative on Trust in IT Systems under the contract 813701 and was performed at the Research Center Secure Business Austria funded by the Federal Ministry of Economics and Labor of the Republic of Austria (BMWA) and the City of Vienna.

References

1. BERR. 2008 information security breaches survey. Technical report, Department for Business Enterprise and Regulatory Reform (BERR) (April 2008)
2. Brank, J., Grobelnik, M., Mladenić, D.: A survey of ontology evaluation techniques. In: SIKDD 2005 at Multiconference IS 2005 (2005)
3. Brewster, C., Alani, H., Dasmahapatra, S., Wilks, Y.: Data driven ontology evaluation. In: International Conference on Language Resources and Evaluation (2004)
4. BSI. IT Grundschutz Manual (2004)

5. DCSSI. Expression des Besoins et Identification des Objectifs de Sécurité (EBIOS) - Section 2 - Approach. General Secretariat of National Defence Central Information Systems Security Division (DCSSI) (February 2004)
6. Fenz, S., Ekelhart, A.: Formalizing information security knowledge. In: ASIACCS 2009: Proceedings of the 2009 ACM symposium on Information, computer and communications security. ACM, New York (2009)
7. Herzog, A., Shahmehri, N., Duma, C.: An ontology of information security. International Journal of Information Security and Privacy 1(4), 1–23 (2007)
8. NIST. An Introduction to Computer Security - The NIST Handbook. Technical report, NIST (National Institute of Standards and Technology), Special Publication 800-12 (October 1995)
9. Patel, C., Supekar, K., Lee, Y., Park, E.: Ontokhoj: a semantic web portal for ontology searching, ranking and classification. In: WIDM 2003: Proceedings of the 5th ACM international workshop on Web information and data management, pp. 58–61. ACM Press, New York (2003)
10. PITAC. Cyber security: A crisis of prioritization - report to the president. Technical report, President's Information Technology Advisory Committee (February 2005)
11. PWC. 2006 information security breaches survey 2006. Technical report, PriceWaterhouseCoopers (2006)
12. Schumacher, M.: Security Engineering with Patterns - Origins, Theoretical Model, and New Applications. Springer, Heidelberg (2003)
13. Uschold, M., Grüninger, M.: Ontologies: Principles, methods and applications. Knowledge Engineering Review 11(2), 93–155 (1996)
14. W3C. OWL - web ontology language (February 2004)

Supporting the Comparison of Business-Level Security Requirements within Cross-Enterprise Service Development

Jason R.C. Nurse and Jane E. Sinclair

University of Warwick, UK
{jnurse,jane.sinclair}@dcs.warwick.ac.uk

Abstract. For businesses planning interactions online, particularly those using Web services, managing risks and accommodating each other's varying business-level security requirements is a complex but critical issue during development. Literature suggests numerous reasons that prohibit the simplistic adoption, or even comparison of requirements; examples apparent in the format used to express them, and processes employed to determine them. This paper presents the initial steps of an approach to ease this process, specially within the context of our cross-enterprise development methodology, BOF4WSS. Specifically, we focus on the design of an ontology to model key factors which influence requirement determination. This ontology will act as the basis for a future tool to state requirements and factors which influenced them, in a common, formal format, to allow for easier analysis and comparison across enterprises.

Keywords: Web services, business-level security requirements, risk management, ontology, systems development approach.

1 Introduction

As Internet usage for business matures, enabling technologies such as Web services (WS) are increasingly being adopted due to the streamlined cross-enterprise interactions they facilitate. In addition to the benefits available however, WS use brings new threats, vulnerabilities and risks, thus making the protection and security of these services a critical consideration. In response to the challenges accompanying WS, there has been a plethora of proposals spanning technical approaches (e.g. standards [1]) to new development methodologies (e.g. [2]). The problem with these proposals however, is that they often tend to be either (i) overly reliant on technology, alluding to standards and systems as the complete solution to WS security, or (ii) overly isolated i.e. focusing on the process one company should follow to secure itself internally, therefore ignoring the broader security issue (discussed in [3]) introduced by WS use.

One approach to addressing these problems is to provide a model for a co-operative development process involving multiple collaborators and in which security issues are a central concern. In previous work we have presented such a

W. Abramowicz (Ed.): BIS 2009, LNBIP 21, pp. 61–72, 2009.

cross-enterprise WS development methodology called Business-Oriented Framework for enhancing Web Services Security for e-business (BOF4WSS) [4]. This methodology builds on the advances of typical systems development models, providing an expanded formalization which can accommodate multiple companies working together.

Each stage of the proposed framework sets out the expected inputs and outputs and the interplay of tasks to be undertaken separately by individual organizations, and those for which joint action is required. Beyond the guidance given, our research has continued to investigate supporting the methodology's stages, and transitions between them. It is this investigation that constitutes the core focus of this paper as we move from the general methodology to concentrate on two specific stages within it—namely the Requirements Elicitation and Negotiations stages—and the interface between them. These stages are of interest primarily because of the inherent difficulty in attempting to compare the high-level security requirements of each business, as they are passed from the individually undertaken Requirements stage into the subsequent joint Negotiations stage. The main problem is directly caused by the possibly disparate nature of security requirements supplied by each business. This disparity is apparent in even basic aspects such as requirements format (natural language can be ambiguous), stated priorities (i.e. importance of requirements to companies) and prioritization schemes (e.g. one business may use high, medium, low, whilst another uses very important, important, less important, not important) used.

The core aim of this paper therefore, is to describe the initial steps of an approach to support stage transition. Specifically we critically analyze security requirement determination methods, to identify factors that influence end requirements, and thus add to the aforementioned disparity. These factors and their relationships are then modelled in a high-level risk-based ontology, which along with the preceding analysis constitute the main contributions of this paper. The motivation behind ontology use is in its inherent suitability to provide a model of the requirements domain that could span, and be communicated across enterprises. The ontology will later also form the basis for an envisaged system to formally express companies' requirements and the factors which lead to them, and then allow for their tool-supported comparison.

This paper is presented as follows; Section 2 outlines the BOF4WSS to provide the context for the current research. In Section 3, we critically examine how security requirements are determined and the most pertinent factors that influence their derivation. Next, the related work and ontology design are presented in Section 4. Conclusions and areas for future work are discussed in Section 5.

2 A Development Process for Cross-Enterprise Security

Like many development processes, our framework, BOF4WSS [4], consists of various stages from requirements gathering, to maintenance. The prime difference is its emphasis on the cross-enterprise development of a multilayered solution, which compiles existing methods, and technologies, into a coherent, well-defined process, to foster high levels of security and business partner trust. In the framework we provide a detailed step-by-step progression with inputs and outputs to stages, whilst identifying what companies do individually, and jointly, and how these two tasks interface. Below, we very briefly revisit the methodology's stages.

The first stage is the **Requirements Elicitation phase** and within it each company works largely individually, analyzing business goals, constraints, organizational security polices, risks and so on to determine their high-level requirements (functional, security and otherwise) for the envisioned WS business scenario. Next, both companies meet in the **Negotiations phase** to compare and discuss requirements, and chart an agreed path forward, especially with regards to the varying expectations likely maintained by each company towards security. The **Agreements phase** which follows, builds on negotiations and suggests a contract to solidify understanding of the requirements thus far, and the definition of an interaction security strategy (ISS), which is a less rigid management structure specifying cross-enterprise security directives to guide interactions. In the following **Analysis/Architectural phase**, companies use the agreed requirements and jointly define conceptual business process models for the envisioned interactions. ISS directives (policies, best practices) are then applied to secure the models. After formal process definition, the framework advocates another **Agreements phase** constituting a thorough legal contract reflecting detailed expectations of parties. The **Design phase** next aims to help businesses define a low-level systems view of how the conceptual vision will be satisfied. This includes identification of relevant standards, and trade-off analysis of their use.

Transitioning from low-level processes and services, the following phase, **Agreements**, focuses on the quality-of-service level, and specifying a mutual understanding of the needs, and guarantees expected by each company for the actual Web services. Aside from formal natural language statements, this specification is done using relevant policy, and service agreement standards. The penultimate stage is the actual **Development & Testing phase** and is largely—but not exclusively—carried out by companies individually, using the agreed systems design specifications and the service agreements, to build and test the services. To aid in this internal process, the framework builds on current research such as [2], but especially stresses the joint verification of planned security measures between companies. Last is the **Maintenance phase** which, apart from encompassing general system updates, stresses the on-going updating and enforcement of security measures in both the ISS and developed systems. This is imperative to ensure that implemented levels of security are preserved.

As mentioned in Section 1, our research has progressed from the sole provision of guidance in the BOF4WSS, to investigate further into developing its stages, and providing systems support wherever necessary to aid in framework use. Of particular interest in this paper is the interface between the Requirements Elicitation and Negotiation phases, due to the inherent difficulties when attempting to compare and negotiate on the high-level security requirements supplied by businesses. In the following section therefore, we present initial steps to solving this problem by analyzing the requirement concept. Our analysis consists of defining a security requirement in the context of this research; briefly discussing methods used to determine requirements; and finally, identifying key factors that influence how companies derive their requirements. These factors are then carried forward to Section 4 where they are used to design an ontology to support stage transition. This general process of considering ontology goals, defining core terms/factors, and identifying relationships, amongst other things, is presented in [5] as a main methodology for the creation of an ontology.

3 Analyzing Security Requirements

3.1 Security Requirement Definition

The notion of security requirements has occupied discussion in countless fields spanning both industry and academia [6,7]. As can be expected therefore, there is an array of definitions for this term, each with respect to the applied domain (e.g. Information Technology (IT), or business) and also, each updated to account for evolutions in thinking. Researchers in [7] illustrate this as they analyze the definitions available, and the numerous ways requirements are regarded in the literature. Common definitions in the IT field ranging from high-level goals, to system-specific mandates—the latter constituting the bulk of the literature.

In this paper as with the framework's stages under investigation however, the term security requirement is aimed at higher levels of abstraction, specifically connoting a high-to-medium level need, largely in terms of the information security goals of confidentiality, integrity, availability and accountability. Requirements thus usually expressed for a general business process or action (hence *business-level* security requirements), as opposed to technical-level constraints. With the definition clarified, we now focus on how requirements are determined.

3.2 Risk-Related Methodologies

Regardless of the intended purpose, a security requirement represents the culmination of a detailed and often non-trivial process. Firesmith [8] researches specifically into the topic of engineering security requirements, and some of the most crucial precursors to defining requirements he identifies are: (i) the identification of assets and services to be protected; (ii) analysis of their innate vulnerabilities; and (iii) a review of the security threats which endanger the assets. Only after this process, can appropriate and ultimately useful requirements be ascertained.

To consider this process formally, and in a comprehensive, organizational context, the series of tasks presented above constitute a process analogous to that of Risk Management (RM). RM as defined by International Organization for Standardization (ISO), representing the full complement of coordinated activities to direct and control an organization with regard to the risks it faces [9]. Reflecting on our comparison of RM to Firesmith's process, the activities to determine risks in RM are similar to Firesmith's three requirement precursors (i.e. (i)-(iii) above), whereas the activities to control risks in RM, match Firesmith's derived security requirements. The strong association of a risk, with assets, threats and vulnerabilities is not new and is seen in various articles such as [6,10].

To aid the complex process of managing organizational security risks, an abundance of methodologies have been offered. These proposals target not only RM in its entirety, but also inclusive tasks such as risk analysis, which for our purposes is defined as "the systematic use of information to identify sources and to estimate the risk" [9]. Following on from this definition, key intentions of the analysis therefore include the identification and assessment of risks, plus their subsequent valuation [6,11,12]. Some popular and well-documented methodologies for RM are the NIST Risk Management Guide [13] and OCTAVE [14], whereas CORAS [15], ISRAM [16] and the IS risk analysis based on a business model [11], provide examples of risk analysis processes.

The wide variety of methodologies available to manage, assess and treat risks is undoubtedly of great use to businesses. By having numerous choices, organizations can adopt one, or a combination of methods that best addresses their needs, and fits their respective company structure and culture. This is with full appreciation that different methods will have varying uses, strengths and weaknesses, and may stress a dissimilar set of aspects. If we consider OCTAVE for example, its creators identify it as being targeted at organizational risk and focused on strategic, practice-related issues [14]; this is as opposed to a technology-focused assessment mechanism targeted at technological risk as in [13].

A similar scenario is apparent in the comparison of risk analysis methodologies when assessing the way risks are valued and prioritized. Authors in [12] identify this as they compared various risk analysis methodologies. Possibly the best example of the difference in focus can been seen in the formula of risk valuation used by the CORAS and ISRAM processes i.e. CORAS prefers simplicity and thus provides a simple, 'impact and probability' approach to determine loss, whereas ISRAM employs a complicated, all-inclusive formula to value risk, thereby stressing accuracy of valuation over simplicity.

The focal point of the highlighted practices is that with different methodologies, different factors will be stressed, different tools used, and ultimately a wide range of security requirements may be determined. As businesses come together to compare their security requirements in the BOF4WSS's Negotiations stage therefore, there is not only a disparity in the requirements "as is" (as identified in Section 1), but there are likely to be many different factors which influence the derivation of those requirements. This fact should not be ignored as it adds to the disparity problem and prohibits any trivial contrasting of end requirements.

3.3 Factors Influencing Requirement Determination

Having given an overview of how security requirements are determined, this section narrows that scope and investigates into the primary factors which influence that process. To do this, we decompose RM into smaller, more well-defined stages, and then analyze each of the previously mentioned risk methodologies, to determine what factors are entertained during that stage. As ISO [9] has previously clearly outlined stages within RM, their guide is used. In our analysis we focused on the lower level, core processes of source identification (which we consider analogous to risk identification), risk estimation, risk evaluation and risk treatment. For ease of reference, the factors identified are italicized.

Risk Identification is defined as "the process to find, list and characterize elements of risk" [9]. Within the methodologies assessed, the general consensus (shared by CORAS, OCTAVE, the IS risk analysis based on a business model and [10]) as to the factors which help ascertain risks, are *assets*, their *vulnerabilities*, and the *threats* to them. This finding therefore confirming the prior analysis in Section 3.2. In other methods such as the NIST Guide, the concept of risk identification is not as definitive. In their definition of risk, and from their Risk Determination stage however, it is apparent that they heavily associate risk not only with assets, vulnerabilities and threats, but also, with the *likelihood of threat occurrence* and *potential of impact* (if the threat occurs). This is another perspective (more in terms of valuation) in which risks can be initially viewed.

Risk Estimation follows, and is the "process used to assign values to the probability and consequences of a risk" [9]. This stage, also known as risk valuation, is often regarded as one of the most critical in a risk analysis [11]. As implied from the definition, the *probability* of a risk occurring and the *consequence* or *impact* if it materializes, are two crucial factors to estimating the priority or level of a risk. This perspective is supported by the fact that ISRAM, OCTAVE, IS risk analysis based on a business model, CORAS and published texts [10], all employ these factors, even if only as the basis for more complicated (quantitative) estimations. In addition to these two aspects, the NIST approach introduces the concept of *adequacy or effectiveness of current controls*; a control, loosely defined as a risk-reducing measure [13]. That factor is included primarily because if a current control is effective at reducing a risk, then the probability the risk will materialize is lower, hence any risk estimation process should consider this reality. This three-pronged outlook is supported by [17], in his method using probability, potential impact and adequacy of controls to estimate risk.

Thus far, all the methodologies assessed aim to be high-level i.e. focusing on general organizational, or IT risks. As the ultimate aim of identifying these factors for our work is linked to the specific lower-level of WS however, assessing Web application risk approaches is also pertinent. To facilitate this evaluation and identify more influential factors, the DREAD model [18] of risk estimation is employed. DREAD uses the *Damage potential* (how great is the damage if the vulnerability is exploited), *Reproducibility* (how easy is it to reproduce the attack), *Exploitability* (how easy is it to launch an attack), *Affected users* (roughly, how many users are affected), and *Discoverability* (how easy is it to find the vulnerability) factors to assess and ascertain a qualitative estimation for risks. As compared to the previous methodologies, DREAD, albeit targeted at a lower level is not drastically different. The 'damage potential' and 'affected users' aspects for example are simply specific categorizations of an *impact*. Whereas the three other factors can influence the *probability* of a risk occurring. For example, if it is easy for an attacker to discover a vulnerability (Discoverability factor) then probability of occurrence will be higher.

Risk Evaluation's purpose is to compare the estimated risks against a given risk criteria to determine the significance of the risk [9]. As stated by ISO, risk criteria will normally include *associated costs and benefits*, *legal and statutory* *requirements*, *socio-economic aspects*, *concerns of stakeholders* and the *determined risk priorities*. With regards to our chosen methodologies, only CORAS treats risk evaluation as a separate stage, however even in that case it only encourages finalizing the risk values, therefore does not state any specific influencing factors/criteria. In OCTAVE, evaluation is a part of risk analysis, whereas ISRAM and IS risk analysis based on a business model, exclude risk evaluation—possibly because they are strictly risk analysis methods. Lastly, NIST moves from Risk estimation directly to assessing control recommendations to reduce risk levels.

Risk Treatment is the final stage to be examined and entails a "process of selection and implementation of measures to modify risk . . . treatment measures can include avoiding, optimizing, transferring or retaining risk" [9]. At this point, risks have been estimated and evaluated, the next task therefore is deciding how they are to be handled. In making this decision, NIST stresses the consideration of various factors, some of which are *legislation and regulations* (do any laws affect the decision), *organizational policy* (are any policies to be considered), and *cost-benefit analysis* of recommended controls (what is the cost—in terms of financial, operational, feasibility—of treatment compared to benefits and effectiveness of control implementation). To look more closely at the cost-benefit analysis and what is involved, [19] provides an excellent example of how some of the aspects mentioned above influence the determination of the treatments. Key factors identified are *policies*, *regulations*, *risk priorities* and *treatment effects and costs*. Apart from OCTAVE which emphasizes the influence of *stakeholders*, none of the other methodologies specially focus on new risk treatment factors.

At this point we have analyzed security requirements and related literature, and highlighted various factors which influence requirement derivation. These factors feed into Section 4 as we look to create an appropriate ontology design.

4 Ontology Design

4.1 Ontology Use

An ontology can be defined as "a set of assertions that are meant to model some particular domain" [20]. These assertions can range from being either very formal (see [21]) and thus utilize special languages such as Web Ontology Language (OWL), to considerably high-level, and therefore use conceptualizations such as diagrams and object-oriented models [20]. Regardless of the level conceived, a prime use of an ontology is in providing a shared understanding of a domain that can be communicated between people, and application systems [21]. This fact is a key motivation for applying an ontology in our work. Specifically, we use it to produce a high-level model of the most pertinent factors from the RM process which influence security requirement derivation. This model will act as a reference point to communicate the relationships of factors between companies, and also further used as the basis for a tool to help in requirements comparison. Next, we discuss the related work, before presenting our ontology design.

4.2 Existing Security and Risk Ontologies

For the purposes of knowledge sharing and management, [22] provides a good example of an ontology used as a novel strategy to support organizational learning in the complex process of RM. The most relevant aspect of this work is the Software Risk Ontology formulated, which identifies critical constituent concepts such as risk evaluation, impact, probability, and mitigation actions, and the relationships between them when analyzing the RM process.

Tsoumas and Gritzalis [23] also employ an ontology as a knowledge resource for the foundation of their work. They investigate into the topic of security management and providing a framework for reusable security knowledge interoperability, aggregation and reasoning, to exploit security knowledge from a range of useful and diverse resource sets. The ultimate aim is to provide a structured approach to help security experts consider various information sources, and better transition from high-level statements in risk analysis documents and policies, to deployable technical security controls. Specifically, they use a Risk Analysis Security Ontology as a basis for development, and some of its pertinent concepts are: assets, vulnerabilities, impact, security policy, controls and countermeasures.

Within the CORAS project, an ontology has also been defined (see [24]) which pays a special emphasis on risk and security requirements. Unlike the articles above which are geared towards formal ontologies, their aim is to use the high-level ontology to convey meaning and illustrate relations between important risk assessment concepts. Concepts include: risk, security requirements, assets, asset values, security policy, unwanted incidents, consequence, and likelihood.

The last ontology for review is that introduced in [19]. The principal aim here was to illustrate key concepts within their risk assessment framework. Once identified, concepts were then used to specify and compute the Return of Security Investment for the various treatment strategies available. This model's concepts are similar to those previously discussed but its most salient factor is the explicit inclusion of risk level, and constituent elements such as frequency and impact.

The works discussed above provide excellent examples of how ontologies can be employed for various purposes within security and risk-related processes. Considering the suitability of these conceptualizations to the identification of factors influencing security requirements however, none of them alone offers an adequate foundation. In [23] for example, this seems to be reasonably comprehensive, by including from initial risk components to the end treatment, however it does not have risk level/priority, a key aspect as identified in Section 3. The aim in [22] to identify a Software Risk Ontology acts to make it too specific an ontology, therefore even though the authors encompass probability, impact and resulting priority/level, they exclude the basic components that constitute a risk such as assets, vulnerabilities, and threats. Conversely, [22] presents the only ontology with an appreciation for varying approaches (called actions) to address the evaluated risks, these being mitigation and contingency actions.

Comparing the reviewed ontologies with the various stages in the RM process; for risk identification, factors previously outlined such as asset, vulnerability, threat, unwanted incident, likelihood and consequence are generally included in the ontologies. Within risk estimation, only the factors of likelihood (taken as synonymous to frequency and probability), consequence (or impact), and end priority are constantly seen. Whereas for risk evaluation and treatment, none of the ontologies explicitly consider laws, organizational policies, or costs as influential factors in deciding risk treatment specifically. Therefore, even though these ontologies may be good for their purposes, for the reasons outlined, they are inadequate candidates for use in our research. The next section builds on these ontologies, and factors from Section 3 to design a suitable model.

4.3 Risk-Based Ontology and Discussion

With consideration of the ontology creation methodology in [5] and a thorough examination of key factors which influence security requirements, we developed the high-level ontology depicted in Fig. 1.

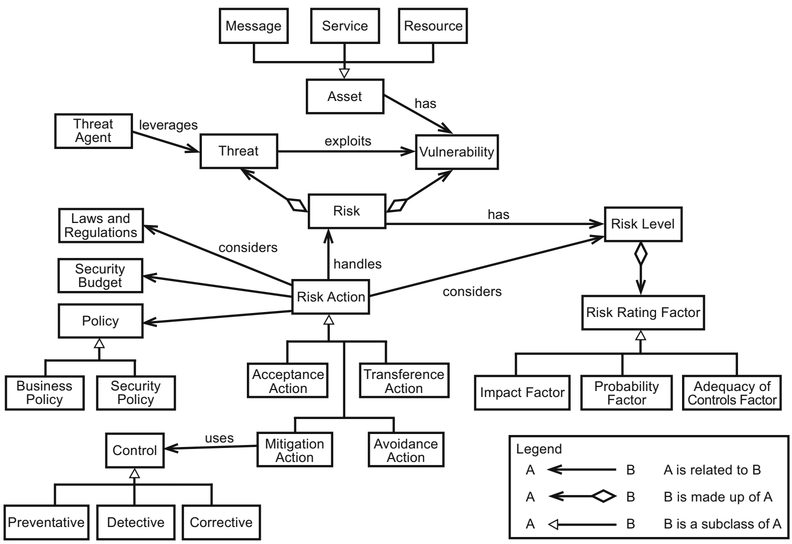


Fig. 1. Risk-based Ontology

In defining this high-level ontology, special emphasis was placed on including factors that were heavily supported in the literature reviewed, and additionally in factors that would be useful when looking at Web application and services security. A Unified Modeling Language (UML)-type notation was preferred to specify this design as it provides a standard, and widely accepted modelling tool to describe concepts and their relations, but also because no standard ontology design notation was identified at the time of writing. Examples of the application of UML to ontology modelling can be seen in [22], and less explicitly in [19].

As a *risk* is the first significant point of contact, our discussion commences there. The identification of a risk typically involves an analysis of the *vulnerabilities* in *assets*, and the *threats* leveraged by *threat agents* to exploit these vulnerabilities. A vulnerability is thus regarded as a weakness; an asset as anything of value to a business; a threat as an undesired event with an adverse impact on an asset; and lastly a threat agent as the cause of the threat [10]. Because our research is interested in WS, it was felt that the notion of an asset could be further decomposed into specific types namely: Web service *message*—an asset in transit from sender to recipient; *service*—the actual Web service application or code; and *resource*—the other assets involved such as data stores, and additional applications and systems.

Regarding risk estimation, the *risk level* concept is explicitly employed to represent the goal of this RM stage. Therefore, a risk after estimation has (or is assigned) a risk level, which in essence is the value or priority grade indicating the severity of the risk. In determining this level, the following risk rating factors were found to be the most pertinent based on their acceptance and suitability for WS: *impact*—the consequence (financial, reputation, clients) if the risk materializes; *probability*—likelihood the risk will occur, based on frequency of past occurrences and subjective factors (e.g. intuition) (probability is estimated without consideration of any controls that might be in place, similar to work in [17]); and *adequacy of controls*—the measures adopted and their effectiveness to mitigate risk [17]. The factors within DREAD were considered but subsequently overlooked due to their close links (discussed prior) with the impact and probability concepts. The theory behind rating factors however, is very open to interpretation, and expression at varying levels of detail. Because of this, future reconsideration of these factors may be warranted. The relationship between the factors chosen and risk level is depicted by the *risk rating factor* concept.

Risk action, the next significant concept, is linked to the risk treatment stage and therefore associated with the handling of risks. The most critical factors considered are *risk level*, *relevant laws and regulations*, the *security budget* (balancing security needs and limited resources is paramount), and *policies* (decomposed into *business*- and *security*-specific). This accounts for the 'considers' relationship from risk action to these concepts. Emerging from risk action itself, specific types of actions are apparent, these are: *acceptance* (if risk level is negligible or cost to mitigate exceeds the asset value); *mitigation* (given a risk is to be mitigated i.e. its level reduced to an acceptable degree); *transference* (if risk is to be assigned to another party e.g. insurance); and *avoidance* (if the risk is handled by eliminating the risk cause e.g. vulnerable asset). This paper specifically notes that the risk action and sub-action concepts are borrowed from work in [22].

The *control* concept next is used by the mitigation action to reduce the identified risk to an acceptable level; a control broadly defined as any risk-reducing measure [13]. Control types found in the literature, and included in our design, are: *preventative*—measures to deter undesirable events from occurring; *detective*—measures that indicate an undesirable event has happened; and *corrective*—measures to recover from the damage caused by undesirable events [10].

As is seen from the designed ontology, the concept of a security requirement—albeit a prime element of this paper—is not emphasized. This is also unlike any of the other conceptualizations discussed [19,24], which all view a security requirement concept as focal to their presentations. The main reason for not directly including this concept in our ontology, is because we view a security requirement as overarching, and more of an end goal. Our aim in designing the ontology was rather focused on logically depicting the most pertinent factors that influence this end goal. Conversely, we do accept that another way this design could be presented is to include the security requirement as a concept between *mitigation action* and the *control*. In this right, a mitigation action would use a *security requirement* to reduce a risk, leaving the control to then implement—to a more specific extent—the security requirement. This way, or the current design both convey logical conceptualizations suitable for the purposes of our work.

5 Conclusions and Future Work

In this paper we critically investigated the security requirement concept, and produced an ontology model of the key factors which influence requirement derivation. The intended usage and novelty of this ontology is in its provision of a shared conceptualization of the requirement domain that could then act as a bridge between companies to reconcile differences in requirement format, determination processes, and so on, at the framework stages highlighted. Beyond this use, the ontology and its preceding discussion also contribute to the general study of risks and security requirements as they critically assess and compare a variety of methodologies, and explicitly identify key factors which determine how organizations decide to handle a risk.

The next step in our research is to use this model to define a formal language. This language will be based on XML, and is the actual tool/mechanism that will enable the formal specification (and later comparison) of the security requirements, risks and the various factors which effect them. The benefits of using the ontology as a base include the fact that it is grounded in accepted literature, and therefore so will be the derived language, and also that it has already identified the main concepts/elements and their relation to each other, a necessary task in defining an XML-based language. Once complete, additional work will research into the ability of the language to encode each business' requirements information, and finally, the evaluation of the comparison tool in supporting negotiations between two companies using BOF4WSS.

References

1. Singhal, A., Winograd, T., Scarfone, K.: Guide to Secure Web Services (SP 800-95). Technical report, National Institute of Standards and Technology (NIST) (2007)
2. Gutiérrez, C., Fernández-Medina, E., Piattini, M.: PWSSec: Process for Web Services Security. In: 4th IEEE International Conference on Web Services, pp. 213–222. IEEE Press, Los Alamitos (2006)
3. Hartman, B., Flinn, D.J., Beznosov, K., Kawamoto, S.: Mastering Web Services Security. Wiley, Indianapolis (2003)

4. Nurse, J.R.C., Sinclair, J.E.: BOF4WSS: A Business-Oriented Framework for Enhancing Web Services Security for e-Business. In: 4th International Conference on Internet and Web Applications and Services (to appear, 2009)
5. King, B.E., Reinold, K.: Finding the Concept, Not Just the Word: A Librarian's Guide to Ontologies and Semantics. Chandos, Oxford (2008)
6. Gerber, M., von Solms, R.: From Risk Analysis to Security Requirements. Computers & Security 20(7), 577–584 (2001)
7. Haley, C.B., Moffett, J.D., Laney, R., Nuseibeh, B.: A Framework for Security Requirements Engineering. In: International Workshop on Software Engineering for Secure Systems, pp. 35–42. ACM, New York (2006)
8. Firesmith, D.G.: Security Use Cases. J. of Object Tech. 2(3), 53–64 (2003)
9. International Organization for Standardization (ISO): ISO/IEC Guide 73 Risk management—Vocabulary—Guidelines for use in standards. Technical report (2002)
10. Landoll, D.J.: The Security Risk Assessment Handbook: A Complete Guide for Performing Security Risk Assessments. Auerbach Publications, Boca Raton (2006)
11. Suh, B., Han, I.: The IS risk analysis based on a business model. Information and Management 41(2), 14–158 (2003)
12. Vorster, A., Labuschange, L.: A New Comparison Framework for Information Security Risk Analysis Methodologies. South African Comp. Journal 37, 98–106 (2006)
13. Stoneburner, G., Goguen, A., Feringa, A.: Risk Management Guide for Information Technology Systems (SP 800-30). Technical report, National Institute of Standards and Technology (NIST) (2002)
14. Alberts, C., Dorofee, A.: Managing Information Security Risks: The OCTAVE Approach. Addison-Wesley, Boston (2003)
15. den Braber, F., Hogganvik, I., Lund, M.S., Stølen, K., Vraalsen, F.: Model-based security analysis in seven steps - a guided tour to the CORAS method. BT Technology Journal 25(1), 101–117 (2007)
16. Karabacak, B., Sogukpinar, I.: ISRAM: Information Security Risk Analysis Method. Computers & Security 24(2), 147–159 (2005)
17. Kairab, S.: A Practical Guide to Security Assessment. Auerbach Publications, Boca Raton (2005)
18. Microsoft Corporation: Improving Web Application Security - Threat Modeling (2003), http://msdn.microsoft.com/en-us/library/aa302419.aspx
19. Houmb, S.H., Georg, G.: The Aspect-Oriented Risk-Driven Development (AORDD) Framework. In: International Conference on Software Development, pp. 81–91. University of Iceland Press (2005)
20. Euzenat, J.: Ontology Matching. Springer, Berlin (2007)
21. Fensel, D.: Ontologies: A Silver Bullet for Knowledge Management and Electronic Commerce, 2nd edn. Springer, Berlin (2004)
22. Falbo, R.A., Ruy, F.B., Bertollo, G., Togneri, D.F.: Learning How to Manage Risks Using Organizational Knowledge. In: Melnik, G., Holz, H. (eds.) LSO 2004. LNCS, vol. 3096, pp. 7–18. Springer, Heidelberg (2004)
23. Tsoumas, B., Gritzalis, D.: Towards an Ontology-based Security Management. In: 20th International Conference on Advanced Information Networking and Applications, pp. 985–992. IEEE Press, Los Alamitos (2006)
24. Gran, B.A., Fredriksen, R., Thunem, A.P.J.: An Approach for Model-Based Risk Assessment. In: Heisel, M., Liggesmeyer, P., Wittmann, S. (eds.) SAFECOMP 2004. LNCS, vol. 3219, pp. 311–324. Springer, Heidelberg (2004)

Federated Product Search with Information Enrichment Using Heterogeneous Sources

Maximilian Walther, Daniel Schuster, and Alexander Schill

TU Dresden, Department of Computer Science, Helmholtzstraße 10,
01062 Dresden, Germany
{walther,schuster,schill}@rn.inf.tu-dresden.de

Abstract. Since the Internet found its way into daily life, placing product information at the user's disposal has become one of its most important tasks. As information sources are very heterogeneous concerning the provider as well as the structure, finding and consolidating the information for generating an all-embracing view on a product has become an important challenge. An auspicious approach for resolving the emerged problems are federated search techniques enriched with Information Extraction and Semantic Web concepts. We investigate possibilities for federated product search combining heterogeneous source types, i.e., structured and semi-structured sources from vendors, producers and third-party information providers. The developed approach is evaluated with a Ruby on Rails implementation.

Keywords: Federated Search, Information Extraction, Ontology, Product Information Management.

1 Introduction

Since the permeation of daily life by the WWW, product information search has become one of the central tasks carried out on the Internet. Producers increasingly realized the importance of presenting their products and associated information in an appealing and informative way online. Additionally, several types of online malls have emerged which today are widely used by consumers as starting points for collecting product information. Due to the fact that providing information on the Internet has become cheap and feasible, also average Internet users have started to enlarge the basis of information by adding so-called user-generated content. As a consequence the Internet now holds many different information sources created by different authors and structured in many different ways. Regarding the amount and the heterogeneity of product information on the Web, the task of information retrieval for one special product involves extensive research including the usage of different vendor, producer and 3rd-party websites.

Unfortunately most sources hold assets and drawbacks considering information quality. For instance, producer websites provide complete and correct information, but use advertising text for promotion purposes. Lexica like Wikipedia contain goal-oriented and fresh information, but are not immune to biased product characterizations.

W. Abramowicz (Ed.): BIS 2009, LNBIP 21, pp. 73–84, 2009.

As per description there are a lot of criteria for information sources to be called ideal. Table 1 presents all conditions that an ideal information source should fulfill.

Table 1. Requirements for an ideal product information source

- Completeness	All available information is included.
- Correctness	All included information is correct.
- Freshness	All included information is up-to-date.
- Neutrality	The information is not biased.
- Goal Orientation	All included information is relevant.
- Comparison	Information from similar products is available.
- Verification	Information is backed by corresponding references.

Obviously no current product information source on the Internet is able to fulfill all of these criteria. Instead a combination of data from many different product information sources could provide an opportunity for satisfying them.

Federated search is the keyword in this context, as it allows the user to gain an integral overview for the product of interest without requesting each information source separately. Additionally, the federation enables access to information sources the user would not have been searching himself. Current providers of applications for simultaneous search already allow the access of structured information sources typically offered by Web Services. However, important additional information for a special product is to be found in semi-structured and unstructured sources like producer websites and websites consisting of user-generated content.

We contribute a reference architecture for the heterogeneous federated consumer product search as well as methods for the extraction of semi-structured information from different information source types. The introduced architecture consolidates information from these sources to offer an all-embracing product view to the user.

The Fedseeko system is a research platform, which implements the described architecture. It uses multiple vendors for building a searchable information basis and enriches the given information automatically by details from other sources, such as producer websites or 3rd-party information providers using information extraction (IE) methods. Experimental results show the feasibility of the developed approach.

2 Related Work

In the field of federated product search, Shopbots [1] emerged already in the mid 90's and were the first step towards integration of multiple vendors in a federated product search using screen scraping. Scraping vendor websites caused a number of problems because it is error-prone and delivers incomplete information. The IPIS system [2] overcomes these problems as it uses Web Service interfaces and ontology mapping as key technologies. The user creates a semantic product query with the help of product categories. Those categories are organized in ontologies to enable an easy mapping between different information providers. As a main drawback, this approach relies on the assumption that each shopping mall has Web Service interfaces and is able to process semantic queries. A more lightweight approach is the shopinfo.xml standard [3]. Shop operators may define a shopinfo.xml that can be downloaded by any shopping portal easily, providing both RESTful Web Services as well as downloading an XML product file for shop federation. Unfortunately, information relevant for buying decisions is not restricted to multiple vendors, but also comprises information on producer websites as well as third-party information, additional data, knowledge or services [4].

The approaches pictured above offer means to consolidate vendor product information, i.e., information provided by online malls like Amazon. Concerning the vendor sources, our approach combines and dilates these works, as we extend the federated product search from integration of different shops with similar interfaces to federation of heterogeneous vendor sources. An adapter-like approach offers the possibility to integrate sources accessible by Web Services as well as Web front ends. Therefore we introduce a generic wrapper adopting specific Web information extraction techniques. Additionally we allow the integration of 3rd-party websites by offering means of information extraction from these sites.

Product information on producer websites often is presented in a semi-structured manner. Extracting and modeling this information is subject to many research approaches as well. Wong et al. [5] describe an unsupervised framework enabling both the extraction and normalization of product attributes from web pages containing semi-structured product information. A number of algorithms are adopted to categorize extracted text fragments and map found attributes to corresponding reference attributes depending on the current product domain. Lermann et al. [6] also picture techniques for extracting semi-structured product information from websites. The developed algorithms are based on AUTOCLASS [7] and ALERGIA [8] and are able to identify page templates and detect table structures to extract information from any semi-structured web page, if only some pages using the same template were given before. Unlike [5], table contents are not directly normalized, that is, the extracted attributes are not matched with reference attributes. Brunner et al. [9] present possibilities to overcome the problem of redundant data management. They describe an architecture for integrating general business object information in ontology entities using different layers of the MOF-model. A highly performant maintenance of ontology information in databases is described as well.

The described approaches offer different possibilities to extract and normalize product information from websites. We introduce alternative methods, especially for the extraction of semi-structured information from producer websites, enabling the extension of the user's product information base with highly relevant and precise information directly from the manufacturer.

The federation of heterogeneous sources using different types of information extraction described in this paper is first presented in [10], providing enrichment of information from online shopping malls with information from product detail pages of producers. We extend this approach to a more generic architecture in this paper. While [10] is restricted to structured vendor information and semi-structured producer information, the work presented in this paper is able to integrate structured as well as semi-structured information from vendors, producers and 3rd-parties in the federated search process.

3 Architecture

Fedseeko's general system architecture is shown in Figure 1. As can be seen on the right side of the figure, Fedseeko is able to query three different types of information sources. These sources are offered by vendors, producers and 3rd-parties. Vendors are online malls like Amazon.com or Buy.com that are able to deliver some basic information about products. Producers are the corresponding manufacturers of the investigated product. They deliver technical specifications of products with a high degree of credibility. 3rd-parties are information providers delivering product information that is often generated by average Internet users. This includes forums, blogs, and test pages. 3rd-party providers also include any dynamic source of product information that is not maintained by vendors or producers, such as search engines. Thus 3rd-parties offer content of varying structure and quality.

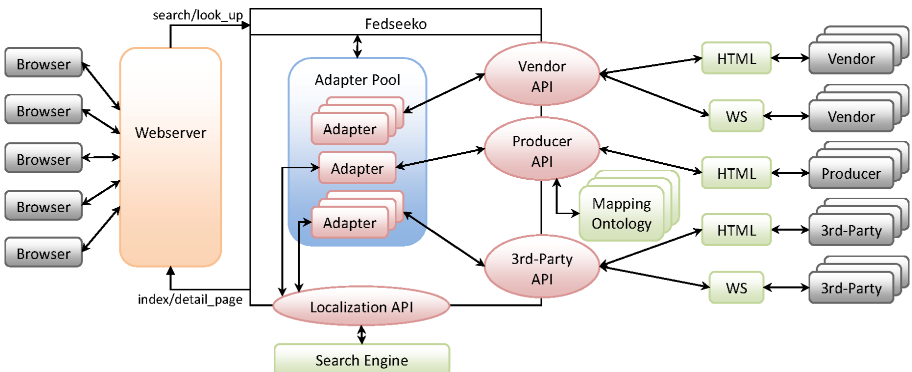


Fig. 1. Architecture of Fedseeko

Consumers access the system through a browser mostly asking for a product list or a product detail page. These requests are executed using the "search" and the "look_up" method respectively. The search method allows providing appropriate parameters, such as a search string, a category, a sort type, etc. The look_up method depends on a corresponding product ID that must be delivered to find the product in the current vendor catalogue. For executing these methods Fedseeko queries one or more adapters that translate the query into a request, which can be understood by the respective vendor.

Using the vendor API the request is sent. When the vendor response arrives, the information is mapped to an internal model and presented to the user. Fedseeko then automatically tries to enrich the vendor information with producer and 3rd-party information. Therefore the Localization API finds out product pages on producer websites using search engines. The information from these pages is extracted, mapped to an internal model using mapping ontologies and added to the user's information base. All tied 3rd-party sources are also queried and available information is presented to the user as well.

The most convenient way for retrieving information is querying Web Services provided by the information sources. As not all sources offer a Web Service, Fedseeko alternatively uses a web scraping wrapper to extract vendor, producer and 3rd-party information. The following chapters will explain the query mechanisms in more detail.

3.1 Integration of Vendor Information

The vendor adapters have to define the before mentioned methods “search” and “look_up” to be included into the system. Then, Fedseeko automatically realizes the integration of a new source and extends the user interface with additional tabs to make the new vendor information accessible. Thus, adding a vendor that provides its product information through a Web Service is made as comfortable as possible, as the adapter is the only thing that has to be created by a user.

An example for a substantial Web Service providing vendor information is the Amazon Associates Web Service (AAWS) [11]. Fedseeko includes the AAWS by providing an adapter that integrates the information through mapping mechanisms allowing the consistent access to product information by controller and model.

Web Scraping Wrapper. For all sources that do not offer Web Services, like Conrad.com or ELV.com, a powerful wrapper was designed that allows Fedseeko to access online malls through web scraping mechanisms. A general purpose adapter for querying these vendor sources offers the functionalities described above. Different parameter values of this special adapter allow the dynamic integration of scraped vendor sources into Fedseeko. These values are saved to the database and consist of details like the structure of the online mall’s URL, the parameters available for the product information source and where to add those parameters to the source URL. The database also holds information about the structure of the result page, which is described by regular expressions, that are better suitable for scraping online malls than XPath-queries, as the HTML-code of online malls is not always 100% clean.

Every user should be able to extend the system’s vendor source pool by adding additional vendor descriptions. As the users cannot be expected to be well versed in the exposure to URL structures and regular expressions, the web interface offers an easy modality to describe the layout of new vendor URLs and result pages. The process is presented in the following.

To create a generic URL for querying a particular vendor, the user has to provide four different values. The first two values consist of URLs generated by the vendor when querying the corresponding web page for a product with a product name consisting of two words, e.g. “ipod nano”. The provided URLs must differ in the page number, e.g. page one and two. The other two values are the used query words, in this case “ipod” and “nano”. Then, by comparing the provided URLs and query words, Fedseeko is able to generate a generic request URL with special flags at all points of interest, being the page spot (where to add the page number), the product spot (where to add the product query) and the separator (used for separating query words in the URL). This generic URL is saved to the database for later use.

The result page is described by regular expressions. To create the regular expressions, the user has to provide some attribute values. These values consist of the attribute name (e.g. “price”), a unique string in front of the attribute value (e.g. “<td class=\"imageColumn\"

width=\"123\">”) and a unique string behind the attribute value (e.g. "<span class=\"aliasName\">"). Additionally the block containing the total set of attributes for one result product has to be described in the same way. Holding these values, Fedseeko creates a set of regular expressions that enable the system to extract all product information from result pages as long as the vendor’s layout is not changed. However, studies showed that the description of vendor layouts by regular expressions makes the system more resistant to layout variations than for instance XPath queries would.

3.2 Integration of Producer Information

After fetching the vendor information from an arbitrary information source, the product can be presented to the user. As the vendor information only consists of few details, Fedseeko is able to enrich this data with details from producer websites.

For dynamically locating additional information sources like product information pages on the websites of corresponding producers, the following algorithm was implemented in the Localization API:

```
producer_page = query(producer_name + " site:com")[0].root_domain
product_page = nil
while(!product_page) do
  product_page = query(product_name + " site:" + producer_page)[0]
  product_name.vary
end
```

In the first step a search engine is queried for the producer’s website by using its name, which is known from the vendor information, and a restriction to the domain “com”. For instance, if searching for a Nikon camera, the query would look like “Nikon site:com”. The first result in this case is “http://www.nikonusa.com”. The system then removes all parts of the URL except the root domain, resulting in “nikonusa.com”. In the second step the search engine is queried for the product name using the previously retrieved producer website to make a hard restriction on the search results. For instance a query for a Nikon camera could look like this: “Nikon D40 6.1MP Digital SLR Camera Kit with 18-55mm site:nikonusa.com”. By this means the system finds the product website among all the producer’s websites. As the product title may not be spelled correctly, Fedseeko tries out different variations of the title elements until it discovers the right page. The producer homepage as well as the product presentation page from the producer are delivered to the user interface.

Now the producer adapter tries to extract product information from a list or table in the product page using the scRUBYt!-API [12]. For being able to do this, corresponding XPath-queries are required. The query set consists of an absolute base query for locating the table in the page and several relative queries for each column of the table. Most producers present their products in a uniform manner. Thus a set of queries is only required once per producer. To generate the XPaths, Fedseeko demands one set of example data for a random product of this producer from the user. For example, if the user examines a Nikon camera, he follows the product page link provided by Fedseeko to go to Nikon’s detail page. There he extracts the items “Image Sensor

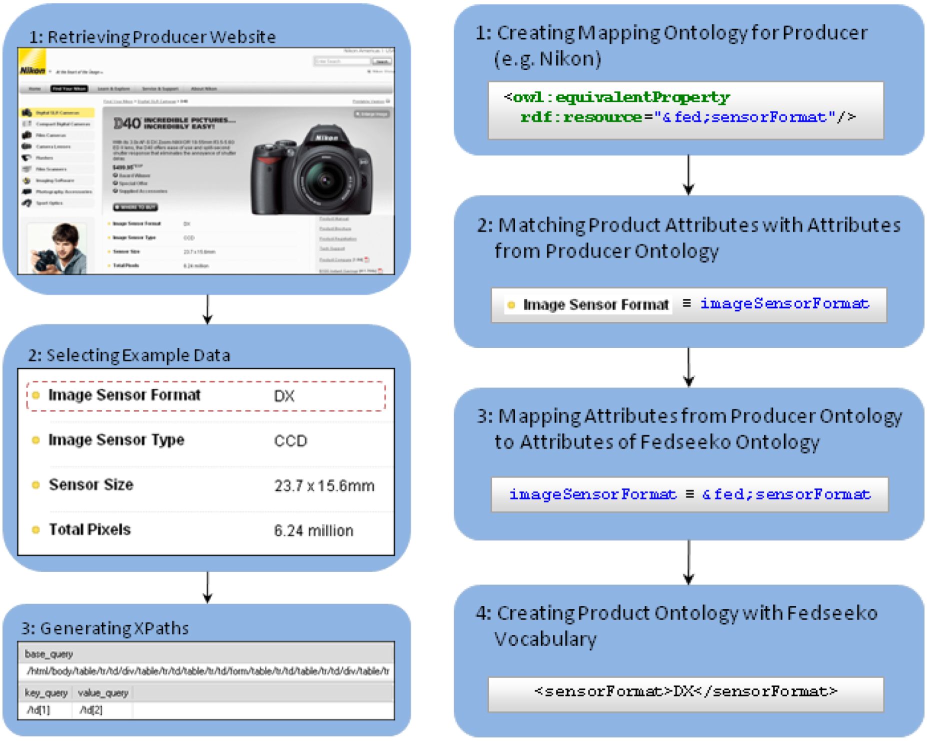


Fig. 2. Extracting and mapping product information

Format" and "DX" from the website and posts this information together with the address of the producer's mapping ontology into a form provided by Fedseeko. Then the system analyzes the page and finds the provided strings.

To avoid problems of different encodings between example data in the website and Fedseeko, the Levenshtein distance [13] is used to compare the extracted data with the different elements of the examined website. After the example data set was found, Fedseeko calculates the corresponding set of XPath-queries and saves them to the local database.

The next time a user investigates a product from Nikon, Fedseeko automatically finds the table information and extracts it. If no table is found (structured product information is often presented in an extra tab of the product page) the system is able to follow some links from the found product page to increase its hit rate. The procedure is displayed on the left side of Figure 2.

For having a consistent view on all product features from different producers, a mapping ontology is used to map Nikon's terminology to Fedseeko's terminology. The ontology holding this mapping information can reside anywhere on the Internet, preferably on the respective producer website. It consists of a taxonomy of product types with corresponding attribute names. Every attribute is extended with a "owl:equivalentProperty"-clause that describes how to map found attribute names to the internal terminology defined by Fedseeko's product ontology. After providing the address of the mapping ontology to the program, Fedseeko translates the producer's attributes into ontology-compliant terms. In a third step the attributes are translated into Fedseeko's terminology and finally used for creating an ontology containing all extracted product information in a consistent format. The described algorithm is shown on the right side of Figure 2. The created product ontology is stored in a public folder to be used by external information systems such as search engines. It is also analyzed by Fedseeko to include the generated product information in the web interface (Figure 3).

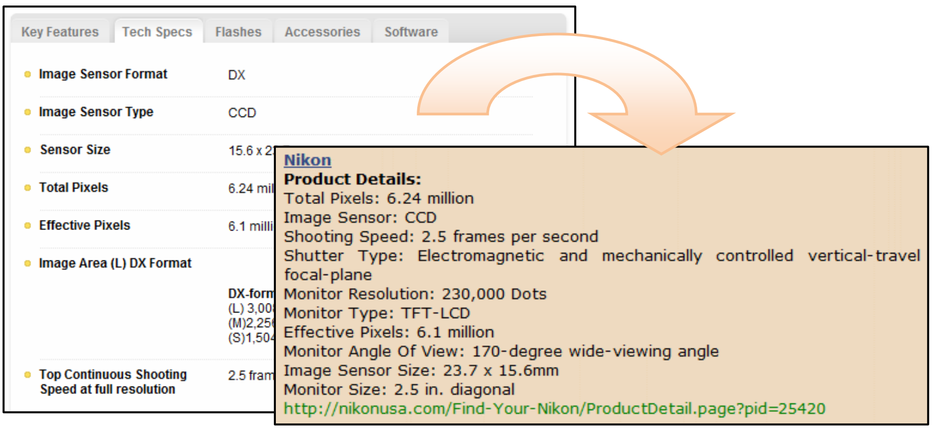


Fig. 3. Product information on nikonusa.com and Fedseeko

3.3 Integration of 3rd-Party Information

The 3rd-party API enables Fedseeko to scrape static websites as well as querying and scraping dynamic websites. Similar to the adapter pattern for vendor information providers, a 3rd-party information adapter can be added for every source to query. The adapter has to provide a method called "query" that accepts a product name, a producer name and a category name, which then can be used in the adapter's sole discretion. The adapter should finally return a list of hashes with the query results that is provided to the view. The view tries to extract a hash value with the key "url" for every list element, which is visually put in the end of each data set. All other hash keys are used as titles for their corresponding values. 3rd-party information adapters can make use of the web form poster, which is a component to automatize the retrieval of information from dynamic web pages. Like the producer information adapter, the web form poster uses scRUBYt! to enable efficient web scraping.

As an example, TextRunner [14] was tied to Fedseeko. TextRunner is a facts-based search engine that is able to deliver assertions stated on different websites all over the Web. These assertions often belong to field reports submitted by users that already possess the product of interest. Thus, Fedseeko uses the adapter to provide the current product's name to TextRunner, which then generates a corresponding list of assertions related to the product. The assertions as well as their source URLs are extracted

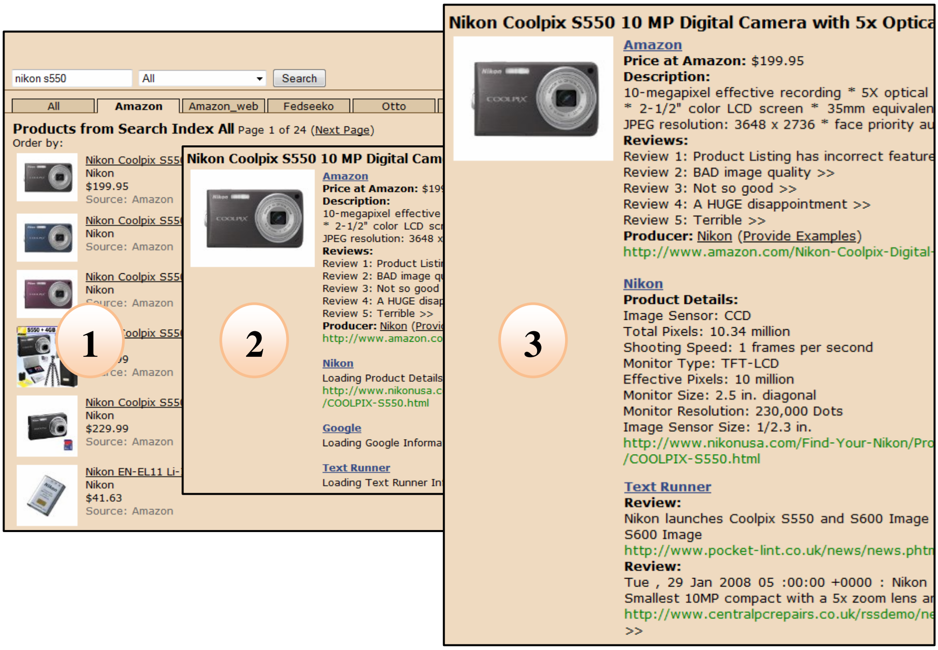


Fig. 4. Screenshot of Fedseeko

and put into the results list. As the TextRunner adapter defines that every contained hash has the two elements "review" and "url", one data set presented in the view consists of a description and a specifically marked source URL (Figure 4).

3.4 Incremental Page Reproduction

Fedseeko was implemented as a prototype providing all features described in this paper. A screenshot of the current system is shown in Figure 4. The figure demonstrates that the different steps of information consolidation are noticeable to the user, because intermediate data is directly inserted into the web interface using Ajax technologies. This way a consumer does not need to wait for the completion of all tasks before examining the additional information.

In Figure 4 five vendor adapters are plugged into Fedseeko, which caused the system to generate five tabs for accessing each vendor respectively. Additionally the all-tab offers the possibility to query all vendor information providers simultaneously. The final product detail page shows producer information, which was extracted from Nikon's website using the generated XPath queries. Below are additional information snippets from 3rd-party sources. Like the vendor adapters, Fedseeko automatically detected all 3rd-party adapters and queried them for product information.

4 Evaluation

To get an idea of the system performance, the success rate of the information extraction from producer websites was evaluated (Table 2). To generate significant test results, Fedseeko was used to query over one hundred products from the Amazon catalogue. A gold standard was created for the whole product set to be able to benchmark the results generated by the system. The gold standard consisted of sets each containing a product name, the corresponding producer name, the address of the producer's website, the websites presenting the product (if available) and a flag describing the existence of a semi-structured element on this website presenting product details. For instance, if talking about digital cameras, a gold standard set consists of the product name "D40", the producer name "Nikon", Nikon's website "http://www.nikonusa.com", the product detail page address "http://www.nikonusa.com/Find-Your-Nikon/ProductDetail.page?pid=25420" and a flag set to "true", as the page contains a table presenting technical information about the D40. The categories analyzed in this evaluation were "Technical" (e.g. Digital Cameras), "Leisure" (e.g. Bicycles), "Body Care" (e.g. Shampoos) and "Media" (e.g. Books). All information was gathered by hand. Then we queried Fedseeko for each of the products and checked, if the gold standard could be fulfilled. The implemented algorithms showed good results, as 71% of existing product information tables could be found and analyzed. Details of the evaluation are presented below.

Table 2. System performance concerning product information

	a_{page}	$r_{localize}$	a_{table}	$r_{extract}$		r_{total}	$r_{localize}*r_{extract}$
Technical	93%	82%	89%	84%		58%	69%
Leisure	83%	76%	68%	94%		41%	71%
Body Care	99%	70%	40%	100%		28%	70%
Media	75%	87%	40%	100%		26%	87%
All	88%	78%	64%	91%		40%	71%

Following data was evaluated: the product page availability a_{page}, which describes how many products have a presentation page; the localization recall $r_{localize}$, which describes how many of the existing product pages were found by Fedseeko; the product table availability a_{table}, which describes how many of the existing product pages own semi-structured information; the extraction recall $r_{extract}$, which describes how many of the existing information tables the system was able to analyze.

For system users it is interesting to know, in how many cases the system is actually able to enhance the product information basis with information extracted from producer pages. Thus, the total recall r_{total} of the extraction procedure was calculated using the following formula:

$$r_{total} = a_{page} * r_{localize} * a_{table} * r_{extract} \,. \tag{1}$$

Obviously technical products are most appropriate for extending their information base with the Fedseeko system (r_{total} = 58%), while only 40% of randomly chosen products can be enriched with producer information. This is mainly rooted in the high amount of available product pages and semi-structured data for products of this kind. Hence the system performance would receive a strong boost, if more producers offered product pages and product information tables. This is proven by the value $r_{localize}*r_{extract}$, which describes the system performance independent from the immutable values a_{page} and a_{table}. Here Fedseeko shows a noticeable performance, as more than $^2/_3$ (71%) of available producer information is found and extracted.

The remaining errors (29%) are caused by the system's algorithms. For example, the product's producer website is not always localized correctly. This especially happens when retrieving web pages of relatively unknown producers or companies with ambiguous names. In this case the product page cannot be localized as well. If the producer's website could be localized correctly, sometimes the localization of the product page still fails, as other websites on the producer's domain may contain the product title and are ranked higher by the queried search engine. Improvements in the localization algorithm would meliorate the success rates considerably as the extraction process already shows excellent results.

Nonetheless the evaluation shows that the overall recall is high enough to offer valuable information to the user. Fedseeko is able to facilitate the creation of an all-embracing view on a product of interest and thus supports the user in taking his buying decision.

5 Conclusions

We have investigated an approach for federating multiple resources of disparately structured types for consolidating product information from all over the Web. Design patterns for integrating vendor information sources of different kinds were shown as well as methods for the dynamic extension of this information by finding and querying information sources from producers and 3rd-parties at runtime to create an all-embracing view for the user. Evaluation showed the success of the approach.

Considering the criteria for an ideal information source mentioned in the beginning of this paper, we have facilitated a noticeable improvement in the field of product information search. The system is not yet delivering complete information about any product of interest, but it collects a high amount of information from different sources. The collected information can be seen as correct in the sense that enough sources are queried to enable the user of the system to compare the retrieved information and filter out conflicts. The information is fresh, as all sources are queried at runtime. Still caching functionalities are envisioned for future versions to allow a better performance when many users access the system simultaneously. The information in its whole is neutral, as information snippets from different sources can be compared with each other. The information is goal-oriented, because no advertisement is included in the view. The comparison of viewed products with automatically retrieved alternative products is not possible yet, as the system needs to be able to identify exact product types for offering this feature. Verification in turn is given, because every information snippet is delivered along with its source URL.

Future works should concentrate on the improvement of the localization algorithm to gain a higher recall concerning the producer information. Additionally, the algorithm for extracting information from product pages should be further automatized. Another important feature would be the personalization of Fedseeko to increase the usability of the system. At the moment an arbitrary amount of information sources can be offered to the user, which could overcharge his receptivity. Instead, the system should suggest some relevant information sources to first-time users and provide the possibility to add and remove information sources in a simple manner for registered users. A fair enhancement would also be the automatic adoption of source suggestions according to recorded usage statistics.

References

1. Fasli, M.: Shopbots: A Syntactic Present, A Semantic Future. IEEE Internet Computing 10(6), 69–75 (2006)
2. Kim, W., Choi, D., Park, S.: Intelligent Product Information Search Framework Based on the Semantic Web. In: 3rd ISWC, Hiroshima, Japan (2004)
3. Wolter, T., Schumacher, J., Matschi, M., et al.: Der shopinfo. XML-Standard. XML-Tage, Berlin (2006)
4. Hepp, M.: The True Complexity of Product Representation in the Semantic Web. In: 14th ECIS, Göteborg, Sweden (2006)
5. Wong, T., Lam, W., Wong, T.: An Unsupervised Framework for Extracting and Normalizing Product Attributes from Multiple Web Sites. In: 31st SIGIR, Singapore City (2008)
6. Lermann, K., Knoblock, C., Minton, S.: Automatic Data Extraction from Lists and Tables in Web Sources. In: IJCAI, Seattle, USA (2001)
7. Cheeseman, P., Stutz, J.: Bayesian Classification (AUTOCLASS): Theory and results. In: Advances in Knowledge Discovery and Data Mining. AAAI/MIT Press, Cambridge (1996)
8. Carrasco, R., Oncina, J.: Learning Stochastic Regular Grammars by Means of a State Merging Method. In: Carrasco, R.C., Oncina, J. (eds.) ICGI 1994. LNCS, vol. 862. Springer, Heidelberg (1994)
9. Brunner, J., Ma, L., Wang, C., et al.: Explorations in the Use if Semantic Web Technologies for Product Information Management. In: WWW, Banff, Canada (2007)
10. Schuster, D., Walther, M., Braun, I.: Towards Federated Product Search From Heterogeneous Sources. In: WWW/Internet, Freiburg, Germany (2008)
11. Amazon: Associates Web Service (2008), `http://www.amazon.com/E-Commerce-Service-AWS-home-page/b?ie=UTF8&node=12738641`
12. Szinek, P.: scRUBYt! - A Simple to Learn and Use, yet Powerful Web Scraping Toolkit Written in Ruby (2008), `http://scrubyt.org`
13. Sulzberger, C.: The Levenshtein-Algorithm (2008), `http://www.levenshtein.net`
14. Banko, M., Cafarella, M., Soderland, S., et al.: Open Information Extraction from the Web. In: IJCAI, Hyderabad, India (2007)

Parallel Algorithm for Query Content Based Webpages Ranking

Jiří Vraný

Institute of New Technologies and Applied Informatics
Faculty of Mechatronics and Interdisciplinary Engineering Studies
Technical University of Liberec
jiri.vrany@tul.cz
http://www.nti.tul.cz

Abstract. The Intelligent Surfer is one of algorithms designed for ranking of search engine results. It is an interesting combination of the PageRank algorithm and the content of web pages. Its main disadvantage is long computation time compared to the PageRank computation time. A computation of the PageRank itself is a very time-consuming process. A lot of papers with topic of efficiency and speed-up of the PageRank computation were published. This paper brings a proposal of speed-up of the Intelligent Surfer algorithm in three steps denoted as the CZDIS algorithm. Experiments with web graph of 1 million nodes size proved that proposed algorithm is usable solution for search engine results ranking in dependance on the page content. Successful implementation of Czech language model experimentally proved possibility of Intelligent Surfer application to different, non English languages.

1 Introduction

Web pages, splitted over a huge number of web servers around the world, are the biggest collections of documents in human history. Moreover, the web environment is ever changing and growing. The search engines return thousands of results for common queries. Functions for sorting search engine results are usually a compound of several algorithms and methods. Nowadays, personalized or content based modifications of traditional algorithms are one of the very popular ways, because they allow adjusting the results depending on users' interests and preferences.

One of the most popular traditional ranking metrics is the PageRank algorithm [2], developed by Larry Page, one of the Google creators. Unlike other algorithms, such as HITS or Hilltop, the PageRank metrics is independent of the page contents, not even of user query. The only indicator of page quality is how many good-quality pages it is linked by. This way of ranking was very innovative in the days of Google creation but nowadays it is not enough. The number of results is very big and more content based ranking algorithms are required.

Lots of papers and algorithms adressing the personalized PageRank were published so far. Research papers such as [8] or [3] describes the most important ones.

W. Abramowicz (Ed.): BIS 2009, LNBIP 21, pp. 85–96, 2009.

The PageRank itself allows possibility of connections between a page or query content and the hyperlink structure of the web. This is possible by using the personalization vector **v** during the computation.

One of the published solutions howto put the PageRank together with content of the document, is the Intelligent Surfer (IS) [1] algorithm developed by M. Richardson and P. Domingos. IS algorithm links together content of the page, user query and the hyperlink structure. The cost of this connection is massive increase of the computation time. The situation is similar with other personalized algorithms. One of the basic principles of personalization of the PageRank is to compute more rank vectors than only one.

In this paper, the Czech Distributed Intelligent Surfer (CZDIS) algorithm is presented. The basic question leading to development of the CZDIS algorithm was if is possibile to use IS based on the English language model for different language and keep its ranking quality. Another focus of the development process was to reduce computation time, the most criticized disadvantage of the IS algorithm. The Czech language model was used during the process.

Theoretical principles of the algorithm were experimentally verified on the web graph with size of 1 millions pages and circa 4 millions of hyperlinks. Results of the experiment are included.

The rest of the paper is organized as follows. Section 2 describes principles of the PageRank and its computation. Section 3 describes the Intelligent Surfer algorithm. Section 4 is the proposal of the CZDIS algorithm, and section 5 provides some details about implementation of this algorithm. Results of the experiments are divided into two parts of the section 6.

2 PageRank Algorithm

Origins of the algorithm are in bibliography metrics and citation counting. As mentioned above, it was invented by Larry Page.

We can say that the web page is good-quality when it is linked by other good-quality pages. We should define the PageRank of a page u as a sum of the PageRanks of linking pages. For a better proportionality only a fraction of PR is given to the linked page. If we label N_u the number of out-links of the page u, then link (u, v) handovers to the page v $Rank = Rank(u)/N_u$.

WWW or its part can be represented as an oriented graph W. Each node of this graph is representing a web page and each edge is representing a web link. Let **M** be the incidence matrix of a web graph W. It applies that $m_{ij} = 1/N_i$, if there is a link from the page i to j, and $m_{ij} = 0$ otherwise. The PageRank vector is the right eigenvector of matrix **M**. But for the provision of eigenvector existence, we have to remove two main problems of matrix **M** - the dangling nodes and the reducibility of the matrix.

If we replace each row representing dangling node by row vector e^T/n, where n is the number of pages, we get a row stochastic matrix **S**. In order to force irreducibility and convergence of computation, we put the teleportation coefficient α to the matrix **S**. We finally get a stochastic and irreducible matrix **G**.

$$\mathbf{G} = \alpha\mathbf{S} + (1-\alpha)\mathbf{E} \quad (1)$$

Where $\mathbf{E} = ev^T$, and v^T is an arbitrary personalization vector. Existence of eigenvector x is covered by the Perron-Frobenius theorem.

Now we can briefly describe the linear system solution method. For further details about the PageRank computation process see [3].

2.1 Sparse Linear System

Computation of the PageRank vector by solving the sparse linear system was suggested by Langville and Meyer in [5]. When the linear system is large and the usage of the direct method is not practicable, a lot of numeric methods can be used. Direct solution of the linear system can be used for a very small sub graphs only. The PageRank vector can be computed by solving the linear system:

$$\mathbf{x}^T(\mathbf{I} - \alpha\mathbf{S}) = (1-\alpha)\mathbf{v}^T \quad (2)$$

and by normalizing solution $\mathbf{x}^T\mathbf{e} = 1$

3 Intelligent Surfer

The keynote of this algorithm is to replace the completely random user by user with some intelligence in his behavior. In the probabilistic model of the intelligent surfer algorithm is random walk replaced by the following idea. The real user doesn't choose the pages randomly, but chooses them according to their content and to content of the search query. Now we take a look at the main principles of the algorithm. The interested reader is suggested to read the original IS paper [1] too.

Let the probability of surfers visit on the page j be denoted $P(j)$. $P(j)$ can be computed as a sum of the teleportation probability and the link transfer probability. This is the same principle as with the original PageRank. In the case of the IS algorithm the probability is expressed in the dependence on query content. Probability distribution over pages is:

$$P_q(j) = (1-\alpha)P'_q(j) + \alpha \sum_{i \in B_i} P_q(i)P_q(i \to j) \quad (3)$$

Where B_i is set of in-links of page i, $P_q(i \to j)$ is the probability of transition to page j, α is the teleportation coefficient and the $P'_q(j)$ is the probability of teleportation (not follow links).

The involvement of a page content is arranged by dividing the web graph W into a large number of smaller overlapping sub graphs. The key for splitting is a dictionary S — a set of all words. For each word $s \in S$ a sub graph is created by the nodes of W which are representing pages containing a word s. For the generated sub graph W_s we compute the PageRank vector in the usual manner.

For creation of the incidence matrix M_s, we need to know the probability of a teleportation and transfer by the link. This probability should be evolved in the dependence on $R_q(j)$ function. $R_q(j)$ is a metric of importance of the term q in the page j:

$$P'_q(j) = \frac{R_q(j)}{\sum_{n \in W} R_q(n)} \qquad P_q(i \to j) = \frac{R_q(j)}{\sum_{n \in F_i} R_q(n)} \tag{4}$$

Where $n \in W$ is the page in the webgraph and F_i is the set of out-links of the page i.

As $R_q(j)$ function we should use various information retrieval metrics. It should be a simple term frequency, the inverse term frequency TFIDF and many others. The simplest variant is to use a binary function with $R_q(j) = 1$ if page j contains term q and 0 otherwise.

The result ranking of IS algorithm is named QD-rank. For the query $Q = [q_1, q_2, \ldots, q_n]$ where q_i is a single term and n is a number of terms in a query, the QD-rank is computed as a linear combination of single rank vectors proper's to each q_i. For better ranking only those documents, which are containing all q_i from Q are included in the final results.

3.1 Scalability

The computation time of the IS algorithm is highly dependable on the size of the dictionary S. If we denote a number of documents containing a term q as d_q, then the sum of d_q for each $q \in S$ is the number of unique document-term pairs denoted as U. The ratio between U and N (the number of nodes in web-graph W) is also a ratio between the computation time of the IS and the PageRank. Authors of the IS claim that the IS requires 165 times more disk space than the PageRank and that the computation of the IS is 124 times slower than the PageRank computation. A better ratio for the computation is explained by faster convergence of smaller graphs and this leads to the computation ratio as $0.75 * U/N$.

4 CZDIS Proposal

The Intelligent Surfer provides good content based ranking function. But the low speed of computation is the principal disadvantage. Can it be speeded up? And, there is another question to answer. Is it possible to use IS algorithm for a different language from English?

The CZDIS algorithm proposal encompasses following modifications:

1. Decrease the U/N ratio by application of language model.
2. Increase the single computation step (computation of one q-vector) by solving a sparse linear system.
3. Solve the sub-graphs in parallel on multiple nodes.

4.1 Decrease the U/N Ratio

With the number of pages in N growing, the number of the terms and thus U also grows. Because of that, the U/N Ratio is not changing much. The only way how to influence the U/N ratio for a lesser computation time is to decrease the number of words in S. This can be achieved by application of a language model and language statistic. In this paper a Czech language model was used.

The first simple step of S reduction is to remove frequently complementary used words – so called stop words. This is a common operation used in many cases of text information retrieval. The first application of the Czech language model – the list of Czech stop words – contains 257 terms and can be found on [9]. Even though the set of stop words is relatively small, there are many pages that contain these words. Stop words are generating large sub graphs and are part and parcel of U. See table 2 in section 6.1.

Second step of S reduction is to replace original terms by their base parts: stems or lemmas. A lemma is the canonical form of the word. In Czech language it is usually the first case of singular. A stem is the part of a word that is common to all its inflected variants. A stem itself can be even a nonsense word. For some words the stem and the lemma are the same. In the Czech language the lemma and the stem are usually different. The Czech language has complex accidence with a lot of exceptions and irregularities. Lemmatization is in practice realized only with a help of a dictionary. On the other hand, a stemming can be realized by algorithm rules. One of functional stemmers for the Czech language was published by L. Dolamic in [6].

The last proposed step leads to the largest reduction of S. As the opposite of the stop words, there is a huge number of unique words. Stop words are generating large subnets and they are not much relevant for result ranking. Unique words are generating very small subnets very often formed by isolated nodes only. Those subnets are not interesting for link analysis algorithms simply because there are no links between nodes. Also good search results for unique words are provided by their uniqueness combined by the common PageRank. There is still very big set of words, that are neither stop nor unique. And for these regular words we need a good quality ranking function.

Frequency of the words in document collection is a power law graph. Reduction of the S by the unique words leads to a significant reduction of the U/N ratio.

4.2 Linear System Computation

As mentioned above and proved in [5], the solving of the sparse linear system is efficient way how to compute a rank vector. For the formulation of a linear system we need to know the personalization vector v_s^T for each $s \in S$ at first. This vector is derived from the probability P' which depends on $R_q(j)$. For the CZDIS algorithm was as a $R_q(j)$ function used the normalized weight of the word w_{sd} defined as:

$$w_{s,d} = fs_{s,d}. \log \frac{|D|}{|s \in D|} \quad (5)$$

Where $fs_{s,d}$ is a frequency of word s in a document d, $|D|$ is the number of all documents and $|s \in D|$ is the number of documents containg word s.

Frequency w_{sd} is more precise weight of term s in a document d than simple binary metrics. The author of the paper thinks, that usage of w_{sd} can lead to better quality of ranking. The test of rank quality with various $R_q(j)$ functions can be topic of a future research.

From the equations 5 and 4 we can evolve relation for v_s^T. Let $v_{si}^T = f_{sd}(i)$ for each $s \in S$. Initial condition of $x^T = 1/|s|$ where $|s|$ is the number of pages containing word s.

4.3 Parallel Computation

Solving of the sparse linear system brings faster computation of a single sub graph. The sum of all sub graph times — sequential computation time — is faster compared with the Power method. But the IS algorithm is a natural parallel problem. It is a classic example of SPMD computation where we repeat the same program on multiple data. A parallel computation of several sub graphs is a unambiguous method to achievement of faster computation time.

Parallel computation is designed as a master-worker algorithm with message passing. The OpenMPI was used as message passing library. It is providing the necessary communication environment and routines for master and worker recovery after a collapse during the computation process.

Decomposition of the problem is dynamic because the size of the generate sub graphs is varying. To minimize I/O operations generated data structures are stored in the master memory and deleted after sending them to workers. Computed vectors are stored by the workers in the master's shared directory.

Master algorithm steps:

1. load inverse index of documents based on the S dictionary
2. generate the first n tasks by using the index, where n is the number of worker nodes
3. assign task to all n available nodes
4. generate next n tasks
5. while the total number of generated tasks is not equal to the size of S, assign the next task to a worker after it has completed the previous task
6. during cycle 5 generate the next x tasks to conserve total number of generated tasks in the memory equal to n.
7. wait for all workers to complete computation
8. terminate the program on all nodes

Worker algorithm steps:

1. as long as the master does not terminate the process do following steps
2. read task from the master
3. solve the linear system
4. write computed vector to the master's directory

5 Implementation

The experimental parallel environment used was a regular PC cluster with 17 nodes — 2x AMD Opteron 252, 4GB RAM each — with Rocks Cluster Linux system. One node served as a master, 16 dual processor nodes were used as worker nodes. That gives total number of 32 workers available for computation. All computers were directly connected to a gigabit Ethernet switch.

The programming environment of the implementation was the Python language and its modules SciPy, Numpy and Pets4py. The PETSc library was used for solving the sparse linear system. This library contains many of sequential and parallel solvers. As an iterative method, GMREs with LU precondition was chosen. This method proved to be most effective in the previous experiment [10]. The coefficient α was set on 0.9 and convergence criteria were set to 10^{-7} as difference between the two last iterative steps.

Basic operations for task construction — creation of incidence matrix and dangling node representig vector were implemented using SciPy. Scientific Python is the Python module for scientific computations. It contains tools for handling sparse matrices and support for MPI usage with Python.

As Python API for PETSc library, pets4py module was used. The PETSc itself is programmed for C and Fortran languages. The Petsc4py was also used for creation of tools for conversion between CSR matrix format (SciPy) and SEQAIJ matrix format (PETSc)

6 Results

Presented results are divided in two sections. First one presents results of the parallel implementation and its performance, speedup and efficiency. The second one presents the results of the ranking function evaluation by a group of volunteers.

6.1 Parallel Performance

Two datasets were used for the experiments. Parameters of them are in table 1. Two dictionaries were generated from both datasets: a full one, containing all words in a dataset and a reduced one, containing only those words that are contained in more than 100 pages. The size of dictionaries and U/N ratio are presented in table 1.

Table 1. Datasets

dataset	*nodes*	*links*	*dictionary size*	*U/N ratio*
tul	45531	150789	315498	150
czdat	998037	3822430	1805572	158

Table 2 presents the reduction of full S by applications of the language model proposed in the section 4.1. Each of the proposed steps is leading to the reduction of S dictionary size. The largest reduction of S is caused by the step 3.

At first, all the vectors for appropriate dictionary were computed on a single node as a sequential algorithm. After that, the computation was run as

Table 2. Aplication of language model

dataset	*S size*	*S100 size*	*U*	*U/N ratio*
tul	315498	9528	6853644	150
tul - stop words	315268	9362	6630472	145
tul stems only	242371	8903	6053120	133
tul_100	8903	8903	4789301	105
czdat	1805572	108422	157689876	158
czdat - stop w.	1805342	106533	152555094	153
czdat stems only	1387907	100538	136080806	137
czdat_100	100538	100538	106512965	107

parallel process on ascending number of workers. The number of workers was in sequence from 2 to 32 with a power of two step (2,4,8,16,32). Computation times presented in table 3 are a aggregated running times of computations, including communication and sub tasks creation time. The speedup ratio is a ratio between time of parallel and sequential computation. As we can see from the figure 1 this ratio is better for the reduced dictionaries than for the full dictionary. That confirms the theoretical premise.

The downtrend of speedup for a higher number of nodes is caused by smaller subsets. There is a big number of them and communications and task creation times are much higher than the computation time itself. This is causing worse load balancing of the algorithm. With a higher number of workers, master node is delayed with task creation and workers are idle waiting for the tasks. On 32 nodes is there still speedup of 23 times but effectiveness of the computation is in downtrend see fig 2. Preparation of sub tasks in advance before computation leads to the better load balancing, but experiments with this algorithm have proved that the total time of the computation is higher than presented and implemented algorithm.

Table 3. Computation time and speedup

	tul_100		tul		czdat_100		czdat	
nodes	speedup	time	speedup	time	speedup	time	speedup	time
1	1,00	4,688	1,00	81,789	1,00	56,131	1,00	939,666
2	1,97	2,380	1,89	43,275	1,91	29,388	1,85	507,928
4	3,82	1,227	3,19	25,639	3,55	15,811	3,04	309,101
8	7,23	0,648	6,87	11,905	7,11	7,895	6,32	148,681
16	14,28	0,328	13,23	6,180	13,94	4,027	12,85	73,126
32	24,89	0,188	22,35	3,659	23,92	2,347	20,48	45,882

For a reduced dictionary, the drop of speedup is lesser than for a full dictionary. However, even for reduced dictionary the speedup did not follow the ideal curve and on a higher number of nodes it starts to sink. The effectiveness of the computation on the *S100_czdat* dictionary is shown on picture 2. The results were computed as a ratio between speedup and a number of nodes.

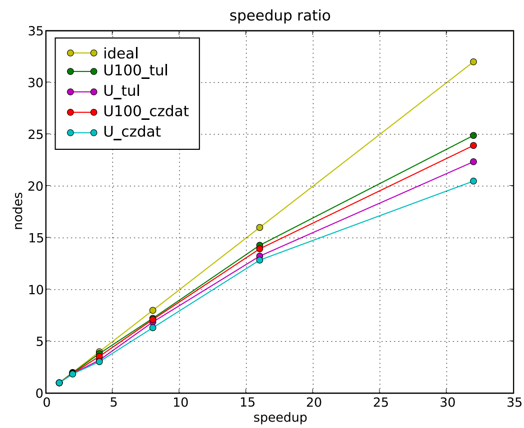


Fig. 1. Speedup of the computation

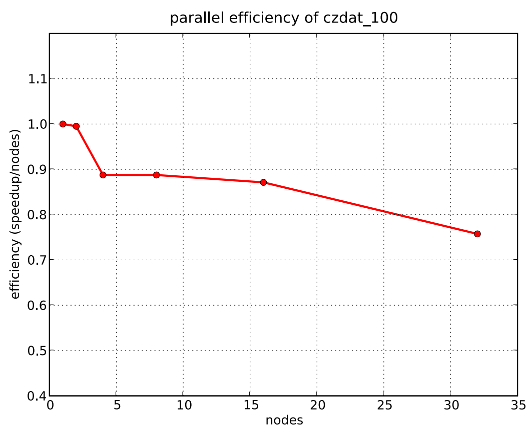


Fig. 2. Efficiency of S100_czdat parallel computation

As we can see in table 4 there are large differences between a theoretical U/N ratio and the real PageRank/CZDIS computation time ratio. The theoretical ratio computed as $0.75 * U/N$ [1] is in range of 79 for the reduced tul dictionary and 102 for the full czdat dictionary. The real ratio is lower in 3 of 4 cases and the computation time is actually faster than a theoretical premise. Only for the non reduced tul dictionary, the ratio is 5 times higher than the theoretical. Why? This is caused by a large difference between the reduced and the full dictionary sizes. A long computation time is caused by the overhead of creation of many small sub graphs. As a matter of interest, the ratio between the CZDIS computation on 32 nodes and the PageRank was computed too. Of course it is a ratio between parallel and sequential computation. For the more accurate results, the sequential computation of the PageRank must be run in parallel first. This experiment is beyond the scope of this paper.

Table 4. PageRank / CZDIS ratio

	tul_100	*czdat_100*	*tul*	*czdat*
pagerank	0,162	14,820	0,162	14,820
single time	4,688	56,131	81,789	939,67
teoretical u/n	79	80	99	102
real ratio	28,936	3,787	504,870	63,41
32 nodes time	0,188	2,347	3,659	45,88
real ratio 32	1,163	0,158	22,589	3,10

6.2 Ranking

For the experiments with the quality of the ranking function, the methodology used by Richardson and Domingos in the original IS paper was used. The number of testing volunteers was increased to 5 (3 in original) and the evaluation scale was extended to five degrees: from 4 for the most relevant document to 0 for completely off topic document.

For the ranking tests, only the *czdat* dataset was used, because the *tul* dataset has shown to bee too small for those tests. It seems that both the CZDIS and the IS works better on a large datasets with high density of links between nodes. The *tul* dataset was useful for the implementation testing but it has too low number of nodes and generated subnets are mostly build from isolated nodes.

The reduced dictionary *S100_czdat* and the according set of vectors was used as a ranking function. The standard PageRank vector was computed too. It was used as a comparative metric and as a ranking function for words not included in used dictionary.

Each tested volunteer firstly suggested two arbitrarily queries. This step created a set of ten random queries. In the next step each volunteer searched

Table 5. Average volunteer grade for each query

czech word (english translate)	CZDIS	PageRank
akademický (academic)	31,2	25,1
aktuálníakce (actual actions)	24,6	28,4
české články (czech articles)	18,2	24,8
financování (funding)	28,4	16,8
jazyk (language)	20,5	10,2
dopis (letter)	28,4	16,2
pole (array, field)	23,4	14,2
směr (direction)	36,2	30,6
rozvrh (schedule)	27,3	19,8
univerzitní ústav (university institute)	23,2	29,8
aplikace (aplication)	34,2	20,5
číslicový(numeric)	23,4	23,8
kontaminace (contamination)	19,2	19,3
average	26,02	21,5

for each of those queries on the testing search engine. For each query 20 results were returned — ten results with the best CZDIS ranking were merged with ten with the best PageRank. Those results were displayed on a screen in a random order. Each result was checked by a volunteer and graded. Results displayed in table 5 are average sums of the grade for each query and ranking function.

The good results of the original IS algorithm [1] were repeated also for the CZDIS algorithm. It means that with the application of a proper language model, the algorithm is usable for various languages, not only for the English one. However the proof for the Czech language is not universal and for some languages it can be hard to construct the language model.

The consequential effect of the reduced dictionary usage is the testing mechanism of volunteers. For words not included in the reduced dictionary the grade should be approximately the same for both functions. From the randomly chosen queries words "kontaminace" and "číslicový" were not included in reduced dictionary. As we can see the grade for both words is almost the same.

7 Conclusions and Future Work

In this paper, the proposal and the results of the CZDIS algorithm implementation are presented. Successful application of the Czech language model is proving, that the Intelligent Surfer algorithm is usable as a ranking function not only for the English language, but also for some different languages.

As a secondary goal of the work, the speed of the computation process was raised up. The application of the language model leads to an improvement of the computation time, without decreasing the quality of ranking. Parallel computation of generated sub graphs is reducing the computation time to the range of minutes. This massive reduction makes possible use of the CZDIS algorithm as a one of the functions for full text search results ranking.

A few problems remain open for the future work. The first one is an application of a suitable lemmatization tool. The question is, if the costs of the lemmatization process will not slow down the computation. The author assumes that lemmatization is a bit complicated than stemming for the Czech language. On the other hand, usage of the lemmas will probably lead to another reduction of the S dictionary.

To appoint IS algorithm as a language model independent, set of tests for various language models must be completed first. One can assume, that for languages, which are know to be complicated for text mining, the implementation of the IS should be hard or even impossible.

At the present time, the author of this paper is focused on the problem of single the PageRank vector parallel computation. As it was proved by Gleich and Zhukov in [7] parallelization is possible if we can handle the problems of the **G** matrix decompostion and load balancing. Author assumes that parallel computation of a single rank vector will lead to another decrease of computation time.

References

1. Richardson, M., Domingos, P.: The Intelligent Surfer: Probabilistic Combination of Link and Content Information in tPageRank. In: Advances in Neural Information Processing Systems, vol. 14, pp. 1441–1448. MIT Press, Cambridge (2002)
2. Page, L., Brin, S., Motwani, R., Winograd, T.: The Pagerank citation ranking: bringing order to the Web. Stanford University technical report (1999)
3. Langville, A.N., Meyer, C.D.: Deep inside Pagerank. Internet Mathmathematics 1(3), 335–400 (2005)
4. Haveliwala, T.H.: Efficient Computation of PageRank. Stanford University Technical Report (September 1999)
5. Langville, A.N., Meyer, C.D.: A Reordering for the PageRank Problem. SIAM J. Sci. Comput. 27(6), 2112–2120 (2006)
6. Dolamic, L., Savoy, J.: Stemming Approaches for East European Languages. In: Peters, C., Jijkoun, V., Mandl, T., Müller, H., Oard, D.W., Peñas, A., Petras, V., Santos, D. (eds.) CLEF 2007. LNCS, vol. 5152. Springer, Heidelberg (2008)
7. Gleich, D., Zhukov, L., Berkhin, P.: Scalable Computing with Power Law Graphs: Experience with Parallel PageRank, Technical Report, Yahoo! (2005)
8. Haveliwala, T., Kamvar, S., Jeh, G.: An Analytical Comparison of Approaches to Personalizing PageRank, Stanford University Technical Report (July 2003)
9. Savoy, J.: Czech stop words (2008), http://www.unine.ch/info/clef/
10. Vrany, J.: Efective computation of personalized PageRank in Python. In: DIR 2009 (submited, 2009)

STAR:chart – Preserving Data Semantics in Web-Based Applications

Irene Celino[1], Dario Cerizza[1], Francesco Corcoglioniti[1], Alberto Guarino[1], Andrea Turati[1], and Emanuele Della Valle[1,2]

[1] CEFRIEL, Politecnico of Milano, Via Fucini 2, 20133 Milano, Italy
{name.surname}@cefriel.it

[2] Dipartimento di Elettronica e Informazione, Politecnico di Milano, Piazza Leonardo, 32, 20133 Milano, Italy
emanuele.dellavalle@polimi.it

Abstract. Every time a Web-based interface is built upon pre-existing data-sources, the result comes often from a difficult and painful process of negotiation between the data manager – who knows the "semantics" of the data to be exposed, but cannot build the UI on his own – and the Web designer – who cares a lot about the presentation and the rendering, but often is not an expert about the content to be displayed.

Semantic Web technologies, with their ability to deal with data, their meaning and their representation, can be of great help in those scenarios. In this paper, we present our approach that makes joint use of Web usability, Web experience, Interaction Design and Semantic Web technologies to develop a Web-application framework called `STAR:chart`, which helps data managers in painlessly building Web-based interfaces upon their data.

We introduce the basic concepts, ontologies and software artifacts that make our approach concrete, together with some use case scenarios in which we successfully employed the `STAR:chart` framework.

1 Introduction

We definitely live in the era of the exponential grow of data scale [1]. Every company or organization has to deal with large datasets which are strictly related to its business and everyday operations. Moreover, the advent and spread of Web technologies provides a world-scale "repository" for the data storage, thus tempting or forcing organizations to expose their data on the Web.

In presence of pre-existing, large scale data sources, often stored in legacy systems, data managers must face the problem of "exposing", "externalizing" the data contained in those sources in order to let other people or organization access them and take advantage of their content. However, this operation is not always easy and painless: apart from the problems of security, privacy and confidentiality of data, data managers must deal with at least two different challenges. On the one hand, they must be able to *integrate* data coming from different sources, if possible creating cross-links between different pieces of information; on the other hand, they must assure that the new application giving access to those data is

W. Abramowicz (Ed.): BIS 2009, LNBIP 21, pp. 97–108, 2009.

usable from the final user point of view and that it *preserves data semantics*. In this paper, we will mainly refer to this second kind of problem.

Our claim is that Semantic Web technologies can facilitate this process of exposing data on the Web and to preserve data semantics in their presentation on the Web. Our long-time experience in the Semantic Web field made us build a lot of prototypes and demo applications to demonstrate the applicability and the usefulness of those technologies to solve real-world problems and challenges. In this paper we want to share our findings in using Semantic Web technologies to solve the aforementioned problem in both traditional and innovative ways: on the one hand we exploit the recognized ability of Semantic Web technologies to represent data and their meaning and to build human-readable and machine-readable applications, while on the other hand we demonstrate how ontologies and concepts/properties mappings can be successfully employed to formalize and to concretize the "semantics" of Web interaction and Web presentation, thus conducting to a new generation of Web applications that preserve the data meaning.

The remainder of the paper is structured as follows: Section 2 illustrates the scenario in which our approach is framed; Section 3 represents the core of the paper, with the explanation of our approach both at high level (Section 3.1) and in details, with the explanation of the defined ontologies (Section 3.2) and the characteristics of the STAR:chart framework (Section 3.3); Section 4 illustrates two real-world examples of employment of the STAR:chart framework in two different use cases; Section 5 presents some related works, while Section 6 concludes the paper and explains the further steps to improve our approach.

2 Problem Statement and Motivation

Currently the process of building a Web-based interface upon pre-existing data sources is a negotiation process between a data manager – who perfectly knows the meaning of the data, their structure and the way to access them in the repositories where they lay – and a Web developer or Web designer – who is not an expert in the data and cares only about the building of a user-friendly interface with its navigational paths.

Too often this negotiation ends with one of the parties getting closer to the understanding of the other party: either the data managers learn how to design and develop Web interfaces or the Web developers become an expert in the data field in order to understand what contents must be visualized in the final application. In other words, the difficulty lies in the background and understanding *gap* that separates the two parties.

We believe that a better scenario can be envisioned: the data manager should remain the "owner" of the knowledge required to understand the data meaning, but he should be also facilitated in creating a Web interface that satisfy his requirements. He should not be exposed to the problem of deciding about how the Web application is structured (e.g., in terms of areas, pages, sub-pages, menus, etc.), but he should be allowed to express what the purpose is of the application, what data should be presented, what concepts and resources are primary and what are accessory, etc.

Our approach relies on the definition of a set of primitives that are familiar to the data manager as well (and not only to the Web expert), that we call *fruition modalities*: if the data managers define the required fruition modalities that hold for his data, the process of building the Web interface that satisfies those needs becomes easier and (at least partially) can be made automatic, so that the required manual work of the Web developer to adjust the result to the data manager requirements is dramatically reduced.

Our idea is that the data managers express, in a declarative way, what "semantics" the Web application should convey: for example, he should define the central elements to be displayed; for each of them, he should list the identifying properties (used to name resources), the most relevant characteristics (used as "summary" or short description of a resource), the detailed attributes (displayed only if the reader wants to know more about a resource) and the related resources (to be linked together); moreover, for each core concept, the data manager should say what functionalities should be allowed over them (like searching, browsing or editing); and so on.

It is clear that this operation is quite easy for the data manager, since it does not require any particular skill in Web application design or development. Nonetheless, this "declaration of intents" is enough for an intelligent Web application framework to sketch a possible Web-based presentation over the data: each resource will be displayed with its identifying properties as "title" and its relevant characteristics as "summary"; if some detailed attributes are listed, a dedicated page should be created (otherwise only the summary will be displayed in some pages); if some related resources exist, they will be listed as links to other pages. Moreover, if a concept is defined as "searchable", a search box must be present to allow users to look for them; if a resource is marked as "editable", the user should be allowed to modify its content; and so on.

Our approach is based on the idea that the way the fruition modalities are translated into Web terms can be – at least partially – automated, by formalizing, through the use of ontologies, the *presentation and interaction semantics* on the one hand and the *Web design models* (like in model-driven Web design literature, as explained also in the following) on the other hand.

3 Our Proposal and Contribution for a Semantic User Interface

Since 2001, we have been investigating how Semantic Web technologies can improve the development and the user experience of Web applications. In particular, we conceived, designed and developed a framework to build knowledge-intensive information portals – called SOIP-F (Semantic Organization Information Portal Framework) – by employing ontologies and other Semantic Web technologies. The results of our research were published [2,3] within the scientific community and successfully deployed in several occasions (like the projects COCOON[1] and NeP4B[2]).

[1] COCOON (IST FP6-507126) integrated project.

[2] Italian FIRB project RBNE05XYPW NeP4B - Networked Peers for Business.

During the last few years, we refined and enriched the framework, both on the conceptual side and on the technical side. The current result is the STAR:chart framework, which, preserving the basic principles of SOIP-F and building on that experience, is an improved tool to help data managers to disclose their information sources by generating – in an easy and semi-automated way – a (Web) application to search and navigate across resources.

3.1 Overview of Our Approach

The innovation of STAR:chart consists in coupling *data semantics* (expressed through the use of domain-specific ontologies) with *presentation semantics* (expressed in our ontologies). We employ ontologies to formally represent the visualization and fruition of resources on the Web; this is enabled by the various studies in different fields like Web modeling, Web usability, user experience and interaction design. Section 5 lists some of the efforts from which we took inspiration when designing our system.

We developed two ontologies – as explained in the following section – to conceptualize the two points of view of the data manager and the Web developer: the STAR:dust ontology – the *presentation and interaction semantics* ontology – specifies the *fruition modalities* and the different roles the input data should play in the final application; the STAR:ship ontology – the *sitemap specification* ontology – specifies the primitives to model the structure of a Web application (taking inspiration from [4]) and represents the *application ontology* of the STAR:chart framework used at runtime to display the resources structured in Web pages.

Our claim is that, once the data manager expresses in a declarative way the desired role of his data in the final application, by the use of the STAR:dust ontology, the system can semi-automatically translate this declaration into a sitemap specification, expressed in STAR:ship terms. This is possible because each Web page can be divided into its basic elements (named travel objects elsewhere [5]), which are called *widgets* in our terminology; those widgets are on the one hand the realization of the fruition modalities (the various ways a user is able to interact with resources published on the Web like searching, browsing, tagging, etc.) and the building blocks of the sitemap on the other hand.

3.2 The STAR:dust and STAR:ship Ontologies

As introduced in the previous section and visually represented in the following Figure 1, the STAR:chart framework has two main underlying ontologies, namely STAR:dust and STAR:ship. This section is devoted to briefly explain the content and the purpose of those ontologies and the way they are employed in the framework functioning.

The **STAR:dust** ontology [6] is a conceptual model aimed at designing and specifying the navigation, that Web users undertake while surfing through resources. It provides a thorough conceptualization that can be used as *application ontology* (in a Model-driven Architecture approach) by the STAR:chart framework, which is a software tool that supports the navigation and the presentation of resources.

Therefore, the STAR:dust conceptual model specifies the *navigation and presentation semantics*. The resulting ontology, however, is not useful *per se*, but it is used to strongly decouple the editing of contents from their visualization. For example, once we have a domain ontology which describes the information contained in the data source, the data manager can "design" the information visualization by *mapping* between the domain ontology and the STAR:dust ontology. Finally, at runtime STAR:chart, taking as input both the domain knowledge and the mappings, makes lever on the STAR:dust ontology and produces a way to present and navigate across contents.

STAR:dust contains the primitive elements to define:

- the *fruition modalities*, i.e. the possible interactions of the final users with the presented information (search, browsing, detail view, tagging, editing, rating, etc.);
- the *presentation building blocks* as they are seen by a generic widget, i.e. the basic characteristics of data (relevant information, identifying features, graphic representation, spatial coordinates, filterable facets, groupable aspects, etc.);
- the *mapping approach*, i.e. how to express a mapping between a domain ontology and STAR:dust itself.

The data manager configuring the STAR:chart framework to present his data uses the STAR:dust ontology to create the *mapping definition*, the declarative artifact which states the "semantics" of data when they are displayed. In this mapping, the user specifies the role of concepts and properties of the domain ontology with regards to the presentation, by choosing a set of fruition modalities to be applied to information (e.g., the instances of the concept X should be browsable) and by mapping values to their visual appearance. An example of use of the STAR:dust ontology for the mapping definition[3] is offered in Listing 1 using N3 notation. In brief, this mapping says that, for the class `sf:Service` of the domain ontology, its property `sf:hasName` will play a double role in the Web-based presentation of all the instances of the aforementioned class: on the one hand, it will be used as title property (`dc:title`) every time a page will be devoted to that resource description; on the other hand, it is an important characteristic (`sd:RelevantProperty`) that should be presented as "identifying" attribute every time a resource of that type is included in a page.

```
:sampleMapping  a  sd:PresentationMapping ;
        sd:onClass             sf:Service ;
        sd:mappingSource       sf:hasName ;
        sd:mappingDestination  dc:title ;
        sd:mappingDestination  sd:RelevantProperty .
```

Listing 1. Example of mapping with STAR:dust

[3] Prefix `sd` indicates the STAR:dust ontology, prefix `dc` the Dublin Core metadata vocabulary, while prefix `sf` refers to Service-Finder ontology, cf. Section 4.1.

The **STAR:ship** ontology, on the other side, is the conceptual model that defines the structure of a Web application in terms of pages, widgets and related artifacts. Modeling a high-level description of a Web site under various dimensions (i.e., structure, composition, navigation, layout and personalization) is not a new idea. The STAR:ship ontology, as a consequence, builds on the long-term research and know-how in Web engineering; in particular, our conceptual model takes inspiration from the well-known and industrially-exploited WebML language [4], from which it borrows the concepts of area, page, unit and link.

At runtime, apart from the mapping definition, the framework needs a model of the sitemap, i.e. an abstraction of the Web site which contains all the elements that make the framework manage efficiently users' requests. The sitemap is at a lower level of abstraction with regards to the data mapping, it is the artifact needed at runtime. Therefore, when configuring the STAR:chart framework with a specific data source, a *sitemap specification* is needed. This sitemap is expressed with regards to the concepts and properties of the STAR:ship ontology and should be derived by the mapping definition itself as well as from the description of widgets and fruition modalities of the STAR:dust ontology. An example[4] is given in Listing 2; this fragment of sitemap specification says that the instances of the `sf:Service` class must be listed in the `searchResultsPage`; to this end, a `ss:ListWidget` is used and a specific template will be exploited for the visualization. A `ss:ListWidget` is configured to display a list of resources and for each of them its `sd:RelevantPropertys`. This means that, at run time, whenever the framework will be asked to display this `serviceListWidget`, it will query the data source to retrieve the `sd:RelevantPropertys` of each listed instance of the class `sf:Service`, as defined in the mapping definition (so that, for example, property `sf:hasName` will be retrieved, cf. Listing 1).

```
:serviceListWidget  a  ss:ListWidget ;
      ss:onClass       sf:Service ;
      ss:hasName       "service list" ;
      ss:hasParent     :searchResultsPage ;
      ss:hasTemplate   "service_list.ftl" .
```

Listing 2. Example of sitemap specification with STAR:ship

In the current implementation of the STAR:chart framework, however, this automatic generation of the sitemap specification is not yet available and therefore this artifact must be provided by the framework user together with the mapping definition. It is in our research agenda the further investigation on this topic: we will enhance (if needed) the two main ontologies and we will find an automated (or semi-automatic) method to generate the sitemap specification from the existing artifacts, in order to reduce the work of customization of the framework, hence easing its adoption and use.

[4] Prefix `ss` in this case indicates the STAR:ship ontology.

3.3 The STAR:chart Framework

A high-level abstract representation of STAR:chart is offered in Figure 1. The framework is able to perform the following functionalities:

1. accessing the data through an abstraction of *data source*
2. preserving data semantics when "translating" resources expressed in the domain ontology to resources expressed in the presentation ontology
3. generating the appropriate Web pages to navigate across resources

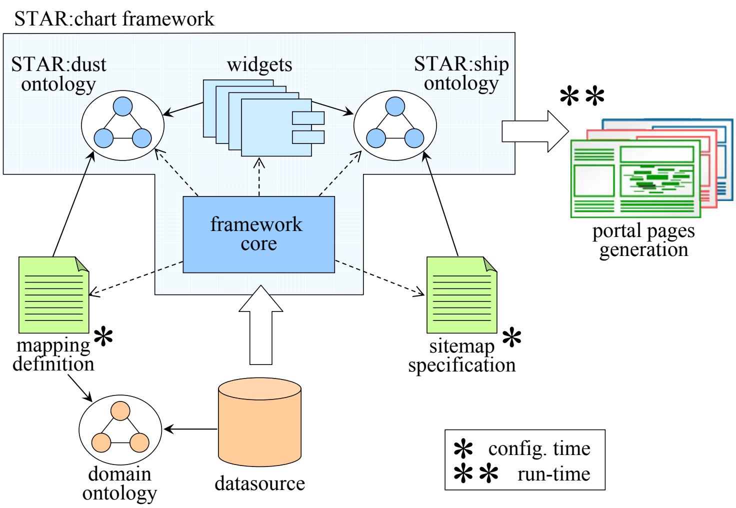


Fig. 1. High-level representation of the STAR:chart framework

Item 1 in the previous list is enabled at configuration time by the connection to the data source and by the "injection" of the domain ontology in the framework. This operation lets STAR:chart access the data abstracting from the specific data repository: the framework uses the data source abstraction to send requests as query-by-criteria[5] and to get the results in RDF format, leaving to the specific data source implementation the (eventual) task of translating queries and converting results. To ease its adoption, the framework provides reference implementations of this data-source abstraction for common sources like RDF repositories.

Item 2 is enabled, at configuration time, by the definition of mappings between the domain ontology and the presentation ontology (STAR:dust). This declarative artifact lets the data manager specify what the purpose is when displaying the information: he can express that some characteristics of data (such

[5] The previous versions of the framework directly submitted SPARQL queries, since they assumed the interaction with a plain RDF store; we extended the system by creating the *Data Source* interface in order to make the framework more generic and able to deal with different kinds of back-end systems. See Section 4.1 for an example.

as properties applied to a concept in the domain ontology) have a specific role when being visualized (e.g., a title, an image, a spatial information) or can be used for performing specific actions (e.g., filtering, searching, comparing, editing). This artifact – the *mapping definition* – is almost the only configuration task requested to the data manager when using the framework. On the basis of this mapping, the most suitable widgets to present the mapped resources can be selected and configured to be used at runtime.

The last item of the previous list, at configuration time, needs the specification of the structure of the sitemap expressed with reference to the STAR:ship ontology (see Section 3.2). In the current stage of the STAR:chart implementation, this *sitemap specification* must be manually provided; our current research, however, focuses on the possibility to automate, at the cost of a loss in expressive power, this specification, by extracting the needed information from the mapping definition (which contains the preferences of the data manager in terms of fruition modalities) and from the STAR:dust and STAR:ship ontologies themselves; the latter ones can formalize some standard compositions of widgets into pages to reflect users' needs and to respect presentation/visualization best practices in terms of usability.

Finally, the generation of Web pages is enabled at runtime by the framework that provides (1) the selection of the suitable widgets on the basis of the sitemap specification, (2) the translation of each widget's information needs in a query over the data source, (3) the access to the data source and (4) the aggregation of the widgets and their rendering in Web pages.

4 Case Studies

In this section, we provide some insights in the use we made of the STAR:chart framework in developing two different applications. The former is related to the design and development of the Web interface of the Service-Finder EU research project, while the latter regards an interface for a logistic scenario as in one of the use cases of the NeP4B project.

4.1 Service-Finder

Service-Finder is an EU co-funded research project, which aims at building a Web 2.0 environment for Web service discovery. Service-Finder will increase the efficiency of service provision and subsequently enable service usage with the ambitious but achievable goal of making the service-oriented-architecture scale to millions of services on the Web. For more information, please visit Service-Finder project homepage: `http://www.service-finder.eu/`

In the context of Service-Finder we employed the STAR:chart framework to design and build the Web-based User Interface, from now on called Service-Finder Portal (SFP). In the following, we briefly explain the configuration steps of the STAR:chart framework we performed to build the Service-Finder Portal on top of it.

The instantiation of the STAR:chart framework in Service-Finder requires:

- the *configuration of the data source*: in the Service-Finder project, the role of data source is played by the Conceptual Indexer and Matcher, which is the component used to store and index all the information about the services. The Conceptual Indexer and Matcher component is based on OntoBroker [7] and Lucene [8]; to this end, an integration API was defined: it expects a query in the STAR:chart framework format and returns results in RDF format;
- the *selection of the fruition modalities* and related characteristics to better present the main objects of Service-Finder (mainly services and providers, with the correlated entities); to this end, we created the mapping definition which contains the various mappings between the Service-Finder ontologies[6] and the STAR:dust ontology;
- the *sitemap specification*, in terms of pages and widgets[7], in order to enable the framework to generate the needed UIs; we designed this sitemap manually, but in the future we will be able to synthetize it starting from the mapping definition and the framework ontologies.

The result can be publicly accessed at `http://demo.service-finder.eu`.

4.2 NeP4B

NeP4B – Networked Peers for e-Business – is an Italian Government co-funded research project aimed at building an infrastructure to help "business peers" to improve and facilitate their business in a coopetitive (cooperative and competitive) scenario. For more information, please visit the NeP4B project homepage at `http://www.dbgroup.unimo.it/nep4b/`

In this context, we employed the STAR:chart framework to build the User Interface of a "peer" in a logistic use case. In this case, the STAR:chart framework acted as a *Semantic Navigation Engine*, to allow browsing and searching through truckload demand and offer. In the logistic use case, different logistic operators expose their data about their vehicles and their planned travels with correspondent goods loads; since those operators are small-medium enterprises, a single "peer" is built by the integration of the data coming from the different actors. The STAR:chart framework is used to build the Web application that gives access to the demand and offer, thus leading to a more effective truckload exchange.

This was made possible by mapping to the STAR:dust ontology the data structures and by defining the search and browse fruition modalities required over those data. The result is a multilingual Web portal on which it is possible to search and browse through load requests and load offers, preserving and

[6] The Service-Finder ontologies are published on the Web, respecting the Linked Data publishing blueprints [9,10], at `http://www.service-finder.eu/ontologies/Service-Ontology` and `http://www.service-finder.eu/ontologies/Service-Categories` respectively.

[7] In Service-Finder, we extended the framework to add Web 2.0-style widgets for collaboration, which previously were not part of the framework.

exploiting the different dimensions of data: the results can be browsed through traditional lists, filtered by faceted browsing and visualized geographically on a map or temporally on a timeline.

For confidentiality reasons of the real logistic operators involved, the public portal available at `http://seip.cefriel.it/truckload-exchange` is only a demonstrative application with faked data.

5 Related Works

As briefly outlined throughout the paper, our work takes inspiration from several approaches in different communities, like Web usability, user experience, interaction design, Web 2.0, etc. In the Semantic Web field, a number of proposals try to address similar problems providing partial or extensive solutions. Without aspiring to being exhaustive, in this section, we provide some pointers to the relevant literature.

The centrality of data meaning preservation is stated also by [11], which argues that, since RDF enables data from multiple sources to be easily integrated to form a single view on a particular topic, traditional Web pages can be considered dead. Semantic Web applications giving access to RDF data (such as Semantic Web browsers) should stop working at the level of pages and start focusing, instead, on the level of "things". This includes supporting different kinds of interactions depending on the object the user is interested in. Available actions and data sources could even change in accordance with the specific task or context, requiring sophisticated models to enable selective integration of heterogeneous sources.

[12] presents Fresnel, an RDF vocabulary which aims at solving the aforementioned issues. Fresnel is based on two concepts: *lenses* and *formats*. Lenses specify which properties of RDF resources are shown and how they are ordered, whereas formats describe how to present the information selected by lenses, optionally adding extra static content. Fresnel's goal is to supply a browser-independent RDF vocabulary which enables users to define how Semantic Web content is presented and is general enough to be applicable across applications.

Designing navigation and presentation tools under the form of "widgets" is the purpose also of Mondeca ITM (Intelligent Topic Manager) with its Semantic Portal and Semantic Widgets [13]. They offer a portal solution based on a widget library: display elements, to be added in a portal, which offer advanced functionalities (natural language requests, search results display and navigation tools, faceted navigation, results maps, subjects relationships mapping). The difference with our STAR:chart approach lies in the abstraction and decoupling layer we inserted through the mapping definition (see Section 3.2) that clearly separates the data layer from the presentation layer.

A similar approach is the one proposed in [14] (and following publications), called OntoViews; it is built upon Apache Cocoon framework and produces valid RDF/XML (instead of plain XML), which is displayed by the use of stylesheets and XSLT. The selection of "views" is however done case-by-case on the basis of the specific data. Another solution, this time based on the Alfresco CMS, is the one developed by [15] for the Cantabria Cultural Heritage Semantic Portal.

All those approaches and solutions can be considered further evolutions of the early works on Semantic Portal such as the well-known SEAL [16], on which is based the website of Institute AIFB. We also owe much to their work; nonetheless, our claim is that we do not only provide a way to build a single Semantic Portal, but a framework to be used to build domain specific Web applications that, as a plus, can be exploited also by data manager without a strong background in Web technologies, helping them preserving their data semantics.

6 Conclusions and Future Work

In this paper, we presented our innovative approach to build Web-based user interfaces and applications to access pre-existing data sources. We explained the problems data managers face today when they decide to expose their data on the Web and the challenges in this context.

We propose a new approach that makes use of ontologies to improve the final application design and development and we introduced the framework we built to demonstrate the feasibility and the goodness of our approach. We also illustrated two real-world applications that successfully make use of the STAR:chart framework.

The main innovation in our work is the employment of formalized "semantics" in the form of ontologies to express not only the meaning of data but also the navigation, presentation and interaction models and their significance in Web design and development. We believe that our approach is not only scientifically and technically sound, but it also has a pragmatic advantage that results in a business value: the agreement between a data manager and a Web designer can be reached easily and painlessly by the use of common abstractions (like the fruition modalities we introduced) that decreases the gap between the two actors points of view. Therefore, the building of a Web interface is less expensive in terms of time and effort.

Our future works, as outlined in the course of the paper, will be devoted to continue our investigation about the automation of the process. The sitemap specification explained in Section 3.2 must be performed manually today, but we believe that it can be largely made automatic by extending the core part of the STAR:chart framework. Moreover, in order to make our solution general and flexible and to fulfill the needs of the largest possible user base, we are continuously adding more predefined widgets to our implementation. The interested reader can keep an eye on our work by following the updates and the new finding at `http://swa.cefriel.it/STAR`

Acknowledgments

This research has been partially supported by the *Service-Finder* EU co-funded project (FP7-IST-215876) and by the *NeP4B* Italian Government co-funded FIRB project (MIUR-2005-RBNE05XYPW).

References

1. VVAA: Big Data. Specials – Nature News 455(7209), 1–50 (2008)
2. Della Valle, E., Brioschi, M.: Toward a framework for semantic organizational information portal. In: Bussler, C.J., Davies, J., Fensel, D., Studer, R. (eds.) ESWS 2004. LNCS, vol. 3053, pp. 402–416. Springer, Heidelberg (2004)
3. Celino, I., Della Valle, E.: Multiple Vehicles for a Semantic Navigation Across Hyper-environments. In: Gómez-Pérez, A., Euzenat, J. (eds.) ESWC 2005. LNCS, vol. 3532, pp. 423–438. Springer, Heidelberg (2005)
4. Ceri, S., Fraternali, P., Bongio, A.: Web Modeling Language (WebML): A Modeling Language for Designing Web Sites. In: Proceedings of the Ninth International World Wide Web Conference, Amsterdam, Netherlands. Elsevier, Amsterdam (2000)
5. Yesilada, Y., Stevens, R., Goble, C.A.: A foundation for tool based mobility support for visually impaired web users. In: WWW, pp. 422–430 (2003)
6. Celino, I., Della Valle, E., Corcoglioniti, F.: The STAR:dust conceptual model. W3C SWD SKOS Use Cases and Requirements Material (2006), `http://www.w3.org/2006/07/SWD/wiki/EucStarDustDetailed`
7. Ontoprise: Ontobroker 5.0. user manual. Technical report (2007), `http://www.ontoprise.de/`
8. Gospodnetic, O., Hatcher, E.: Lucene in action. Manning Publications (2004)
9. Berrueta, D., Phipps, J.: Best Practice Recipes for Publishing RDF Vocabularies – W3C Working Draft (2008), `http://www.w3.org/TR/swbp-vocab-pub/`
10. Bizer, C., Cyganiak, R., Heath, T.: How to Publish Linked Data on the Web (2007), `http://sites.wiwiss.fu-berlin.de/suhl/bizer/pub/LinkedDataTutorial/`
11. Heath, T.: How will we interact with the web of data? IEEE Internet Computing 12(5), 88–91 (2008)
12. Pietriga, E., Bizer, C., Karger, D.R., Lee, R.: Fresnel: A Browser-Independent Presentation Vocabulary for RDF. In: Cruz, I., Decker, S., Allemang, D., Preist, C., Schwabe, D., Mika, P., Uschold, M., Aroyo, L.M. (eds.) ISWC 2006. LNCS, vol. 4273, pp. 158–171. Springer, Heidelberg (2006)
13. Mondeca: Semantic Portal - Semantic Widgets (2008), `http://www.mondeca.com/index.php/en/intelligent_topic_manager/applications/semantic_portal_semantic_widgets`
14. Makelä, E., Hyvönen, E., Saarela, S., Viljanen, K.: OntoView – A Tool for Creating Semantic Web Portals. In: Proceedings of the 3rd International Semantic Web Conference (ISWC 2004), Hiroshima, Japan (2004)
15. Hernandez Carrascal, F., Rodrigo, L., Contreras, J., Carbone, F.: Cantabria Cultural Heritage Semantic Portal (2006), `http://ftp.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-295/paper02.pdf`
16. Maedche, A., Staab, S., Studer, R., Sure, Y., Volz, R.: SEAL - Tying Up Information Integration and Web Site Management by Ontologies. IEEE Data Engineering Bulletin 25(1), 10–17 (2002)

Advanced Social Features in a Recommendation System for Process Modeling

Agnes Koschmider[1], Minseok Song[2], and Hajo A. Reijers[2]

[1] Institute of Applied Informatics and Formal Description Methods
Universität Karlsruhe (TH), Germany
koschmider@aifb.uni-karlsruhe.de

[2] School of Industrial Engineering
Eindhoven University of Technology, The Netherlands
{m.s.song,h.a.reijers}@tue.nl

Abstract. Social software is known to stimulate the exchange and sharing of information among peers. This paper describes how an existing system that supports process builders in completing a business process can be enhanced with various social features. In that way, it is easier for process modeler to become aware of new related content. They can use that content to create, update or extend process models that they are building themselves. The proposed way of achieving this is to allow users to generate and modify personalized views on the social networks they are part of. Furthermore, this paper describes mechanisms for propagating relevant changes between peers in such social networks. The presented work is particularly relevant in the context of enterprises that have already built large repositories of process models.

Keywords: Social Networks, Business Processes, Modeling Support, Personalization.

1 Introduction

Social networking sites such as `Flickr` or `Facebook` enjoys great popularity because they enable to connect and share information with each other and thus support to build online communities of people who have similar interests, or who are interested in exploring the interests of others [1].

An activity that may profit from a transfer of social networking features is *business process modeling*. Earlier, we developed a support system for people involved in this act [2]. The system takes into account a process builder's modeling intention and patterns observed in other users' preferences and uses a repository of process model parts to recommend her efficient ways to complete the model under construction. Recently, we extended this system with some social networking features [3].

Through this extension, process builders can establish who already selected and reused specific process models. Novice process builders in particular can profit from this, as the system encourages user trust and participation. To implement these features, three kinds of social networks have been implemented

W. Abramowicz (Ed.): BIS 2009, LNBIP 21, pp. 109–120, 2009.

in the system: (1) a social network from a *process model repository*, (2) a social network from a *user history* and (3) a social network from an *insertion history*. The social network from the *process model repository* provides an organizational view of business processes. The social network from *user history* shows the relationship among modelers who use the recommendation system. Finally, the social network from *insertion history* shows the relationship among modelers who decided for equal recommendations.

Over the past months, we evaluated the 'socially enhanced' recommendation system in practice, most notably through an experiment at the University of Karlsruhe that involved 30 novice process modelers. These modelers had to build a fairly large process model for which they had all the features of the recommendation system at their disposal, including information on previous usage of process model parts by people being socially close to them. We came to the conclusion that the new social features are not sufficient to propagate trust and thereby to influence novice modelers in their choice of process modeling parts. The main reason we identified is that the current system does not consider *personalized* interests and knowledge of the modelers. In particular, a user can not include her capabilities or working environment characteristics in her network, which makes it difficult for other modelers to evaluate the relevance of her modeling actions. Additionally, we realized that process builders may have an ongoing interest in notifications of changes in process models that are relevant in their social network, i.e. receive and send updates on process models beyond the exact times that they are actually busy modeling.

Against this background, this paper deals with two types of extensions of the social features of our recommendation system:

1. The *personalization of social networks* will stimulate the development of communities of process builders that share the same interests or knowledge, and activate new ways of collaborating. An example scenario would be a group of process modelers that work in different geographical units of a multinational organization, but through the recommendation system work on the modeling of similar procedures in their respective units.
2. The *propagation of changes* will allow users to become aware of potentially relevant updates to the process models they are building (of have built) themselves. Conversely, process builders can actively inform other process builders about relevant changes in process models. In this way, enhancements of process models, e.g. in response to changing legislation of market conditions, will not need to be re-invented over and over again.

To explain how the described extensions are achieved, the following section provides an overview of our metadata model, which lays the foundation for personalized social networks. In Section 3, a method for personalizing the three kinds of social networks will be presented. In Section 4 two algorithms will be presented supporting the propagation of changes within such networks. Section 5 gives an overview on related work. Finally, our paper concludes with a discussion of our approach and an outlook on future research.

2 Metadata Model

To personalize the three kinds of social networks sketched above (social network from a *process model repository*, social network from a *user history* and social network from an *insertion history*) it is necessary to build a metadata model, which describes the skills, and knowledge of users. Based on this metadata model the recommendation system can suggest additional connections to process builders on the basis of common skills. Generally, the interest or the knowledge of process builders is driven by four properties. They are *subject domain*, *capability*, *work environment*, and *individual and group behavior*. Note that users can select predefined properties for the first three properties. Individual and group behavior patterns can be derived from the user history. More specificly the properties are defined as follows:

1. *subject domain* describes the topic the user is working on or has interest (e.g., engineering, medicine, government or manufacture). Process models are usually classified based on their domain and process builders usually have their own subject domains. For example, a process builder who has some previous experience on governmental processes can more easily understand similar processes in government settings. Thus, when a process builder is working on a specific domain, she can consult with the people who have an expertise in it.
2. *capabilities* of process builders include general literacy and technical skills. Also expertise or special skills of users can be described by this property. For example, if a process builder is modeling a quality management process in a software company, people who have deep knowledge of software engineering can help her.
3. *work environment* describes the department/project the user is involved in. The work environment is the functional department or project team to which a process builder belongs. Such properties can normally be derived easily from general users' profiles. And even when a new process builder is added into a social network, those properties can be easily derived from a past track record.
4. *individual and group behavior* patterns and history describes the degree of sociality of users. Note that this property is normally not specified when a process builder becomes active for first time. It is based on the process model design history in the recommendation based support system.

Table 1 shows an example of the user history generated from the modeling history of the community of users. In the table, each row refers to a user (u) and a column corresponds to a process model (p) in the repository. Also, each cell (c_{ij}) shows the number of use of the process model (p_j) by the user (u_i). From this kind of table, we can derive *individual and group behavior patterns* [3]. For example, by applying data mining techniques (e.g. K-means clustering), we can extract several groups in which people have a similar behavior pattern or similar preferences. Arguably, the people who have a similar pattern can provide

Table 1. User History

	P_1	P_2	P_3	P_4	P_5	...	P_N
$user_1$	2	1	0	0	0	..	2
$user_2$	2	1	0	0	0	..	2
:	:	:	:	:	:	:	:
$user_M$	0	0	1	0	0	..	0

more useful information for a process builder. From the above definitions, the metadata model can be defined as follows.

Definition 2.1 (Metadata). A metadata, M, is a 9-tuple $(N, D, C, W, B, P_D, P_C, P_W, P_B)$ where

(i) N is a set of users,
(ii) D is a set of subject domain,
(iii) C is a set of capabilities,
(iv) W is a set of work environment,
(vi) B is a set of behavior patters,
(vii) P_D, P_C, P_W, and P_B is a set of profiles, where $P_D \subseteq (N \times D)$, $P_C \subseteq (N \times C)$, $P_W \subseteq (N \times W)$, $P_B \subseteq (N \times B)$.

For convenience, we define an operation on profiles: $\pi_N(p_D) = n$, $\pi_D(p_D) = d$ for some profile $p_D = (n, d) \in P_D$. We also use $\pi_C(p_C), \pi_W(p_W), \pi_B(p_B)$ in the same manner. For example, there are two users ($user_1$,$user_2$). $User_1$ has a specialism in "government" processes, has a capability of "process design", and belongs to the "IT dept". $User_2$ is working on processes in a "bank", is able to do "programming", and works at the "CS dept". Their metadata is defined as follows.

$N = \{\text{"}user_1\text{"}, \text{"}user_2\text{"}\}$,
$D = \{\text{"}government\text{"}, \text{"}bank\text{"}\}$,
$C = \{\text{"}process\ design\text{"}, \text{"}programming\text{"}\}$,
$W = \{\text{"}IT\ dept\text{"}, \text{"}CS\ dept\text{"}\}$,
$P_D = \{(\text{"}user1\text{"}, \text{"}government\text{"}), (\text{"}user2\text{"}, \text{"}bank\text{"})\}$,
$P_C = \{(\text{"}user1\text{"}, \text{"}process\ design\text{"}), (\text{"}user2\text{"}, \text{"}programming\text{"})\}$,
$P_W = \{(\text{"}user1\text{"}, \text{"}IT\ dept\text{"}), (\text{"}user2\text{"}, \text{"}CS\ dept\text{"})\}$

3 Personalized Social Networks

The metadata model builds upon the three kinds of social networks as mentioned above. We regard these social networks as public, in the sense that any process builder using the recommendation system can access these networks. However, since the potentially huge size of such networks and the risk of information overflow it will be difficult for process builders to retrieve the proper content. Therefore, *personalized* social networks are needed. In this section, we explain how to generate such personalized social networks from the public social networks. First of all, we define a social network as follows.

Definition 3.1 (Social Network). A social network, S, is a 3-tuple (N, A, σ) where

(i) N is a set of nodes,

(ii) $A \subseteq (N \times N)$ is a set of arcs,

(iii) $\sigma : A \rightarrow \mathbb{R}$, is a weight function of an arc.

N refers to a set of users, A shows the relationship between users, and σ is a function indicating the weight of each arc.

To reduce the size of the network, a process builder can consider two kinds of information. Firstly, she can take into account people who she already knows well. In this case, she does not need any metadata of the people, but the name (or ID) of the people would be sufficient to reduce the size of the network. Secondly, she can utilize metadata of people. Based on her own interest, a user can remove some uninteresting nodes from the network and generate a personalized network. To do this, we define a filtering function.

Definition 3.2 (Filtering Function). A filtering function, $F : (S, M, N_1, P_{D1}, P_{C1}, P_{W1}, P_{B1}) \rightarrow S$ where $N_1 \subseteq N, P_{D1} \subseteq P_D, P_{C1} \subseteq P_C, P_{W1} \subseteq P_W, P_{B1} \subseteq P_B$, is defined as follows: $F(S, M, N_1, D_1, C_1, W_1, B_1) = \{(n, a, \sigma) | (n, a, \sigma) \in S \wedge (p_D, p_C, p_W, p_B) \in M \wedge \pi_N(p_D) = n \wedge \pi_N(p_C) = n \wedge \pi_N(p_W) = n \wedge \pi_N(p_B) = n \wedge n \in N_1 \wedge \pi_D(p_D) \in P_{D1} \wedge \pi_C(p_C) \in P_{C1} \wedge \pi_W(p_W) \in P_{W1} \wedge \pi_B(p_B) \in P_{B1}\}$

A public social network (S) and the original metadata (M) are given as inputs of the filtering function. In addition, the function requires filtering options. In the formula, N_1 represents a set of process builders who will be included in the personalized network. It enables a user to include people in whom she is interested in. A user can also utilize metadata. P_{D1}, P_{C1}, P_{W1}, and P_{B1} refer to a set of *subject domain*, *capability*, *work environment*, and *behavior pattern* respectively. They cover metadata which a user wants to contain in the personal network. As a result, the function returns a network in which irrelevant nodes are removed from the original network, such that it can be used as a personalized social network.

Figure 1 shows an example of such personalization. Figure 1(a) shows a public social network which contains 10 users. Table 2 shows a fragment of the metadata of the users. For example, $user_6$ has "government" as a subject domain and her capacity is "programming". She works at "IT dept." and her behavior pattern is not specified. If a user wants to personalize the public network including $user_5$, $user_6$, $user_9$ and $user_{10}$, "government" as a subject domain, and "IT dept." as a work environment, she can use F(SN,M,{$user_5$, $user_6$, $user_9$, $user_{10}$}, {"government"},{*},{"IT dept."}, {*}) and derive the personalized social network in Figure 1(b).

4 Propagation of Changes in Public and Personalized Social Networks

The public and the personalized social networks in Figure 1 are subject to contant dynamic modifications, for example because a user will select new process

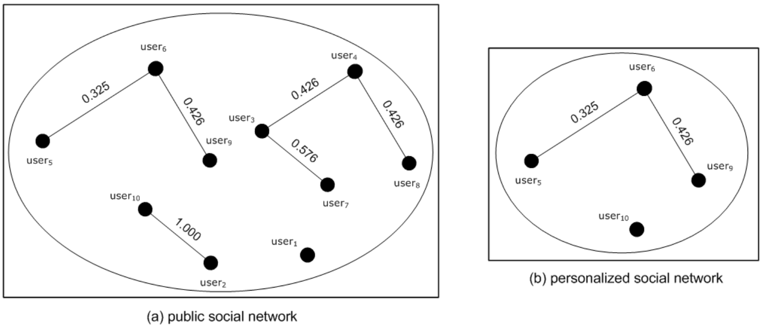


Fig. 1. Social networks

Table 2. Metadata of users

	subject domain	*capability*	*work environment*	*behavior pattern*
:	:	:	:	:
$user_5$	{government, bank}	{process design}	{IT dept.}	{}
$user_6$	{government}	{programming}	{IT dept.}	{}
$user_7$	{bank}	{accounting}	{IT dept.}	{}
:	:	:	:	:
$user_9$	{government}	{accounting}	{IT dept.}	{}
$user_{10}$	{government}	{process design}	{IT dept.}	{}

models in her job. (Note that social networks can change due to structural updates, e.g., when somebody moves from Marketing to the IT department, but these changes are rather straightforward and will not be considered in this paper.) Assume *user 10* starts selecting the same process models as *user 6*. Consequently, over time *user 10* will become connected to *user 7* and thus emulate recommendation patterns of *user 10*. In practice, we realize that process builders want to be informed if they are affected by modifications and users want to notify others if adaptation in their models are required.

We have implemented such change propagations with *pull* and *push* services. *Push* services in the recommendation system involve actively sending (or pushing) information to a specific process builder that the process builder knows to be interested in this info. This requires an active participation of users. In contrast are the *pull* services, which are more passive then the *push* service. A *pull* service involves process builders that specified that they want to receive information if a certain process model has been changed. Pull services will be explained in Section 4.1 and push services in Section 4.2.

4.1 Pull Service

A pull service can be implemented for the propagation of modifications in process models, in model parts or of process elements. In order to consider these three applications of propagation we initially differentiate the types of changes, which may occur in process models, model parts and for process elements. Changes can be of a primitive or of high-level nature, as defined in [4]. High-level changes in our context affect process model parts and are the following:

1. *insert process model part*: a completely new process model part is inserted in a business process model, which has been created by a previous user,
2. *delete process model part*: a process model part has been completely deleted in a business process model.
3. *move process model part*: a process model part has been moved in a business process model.

Primitive changes affect process elements and process models, and pertain to the following: 1)*insert* completely new node and 2)*delete*, 3)*rename* or 4)*move* a node in a certain process model, which is already stored in a process repository.

Primitive and high-level changes affect public and personalized social networks if process builders prefer process models of other and thus emulate their recommendation patterns. The consequence of this amalgamation is that users will be connected to other process builders and thus the social network structure will change. To notify process builders of changes in process models that are relevant in their social network the recommendation system incorporates two algorithms for the calculation of primitive and high-level changes.

To calculate the number of primitive changes in a process model we use the following data structure. We define a class *Node*[1] with the following five attributes, that describe the state of a node: (1) *bInitial* of type *boolean*. If *bInitial* is *false* then a node has been newly inserted. Else no change has been performed, (2) static final *MOVED* of type *int*. This state is valid, if a node has been moved, (3) static final *RENAMED* of type *int*. This state is valid, if a node has been renamed, (4) static final *DELETED* of type int. This state is valid, if a node has been deleted.

To query the state of a node we use the variable *iState* of type *int*. The variable *iCounter* of type *int* is used to count the number of states of a node. The *list nodeList* is used to store all nodes of a business process model. With this data structure we can define an algorithm calculating the number of primitive changes as described in algorithm 1.

This algorithm differentiates between three cases. Case 1 (line 4) is valid, if all elements of a process model stored by a process builder in the repository have been reused by another user without any change. Case 2 (line 7) is valid, if the state of a process model stored in the process repository has been modified by moving, renaming or deleting process elements. In case 3 (line 10) the algorithm counts the number of newly inserted elements and the number of movements, renamings, and deletions of initial available process elements. If the user is interested in the modification of nodes of a specific process model then we prepend to line 1 an if-clause that checks a certain process model name.

[1] The business processes and parts stored in the repository have been modeled with Petri nets. Thus, in this context a node can be a place or a transition.

Algorithm 1. Algorithm to calculate the number of primitive changes

```
int iCounter;
int iState;
for all Node n in nodeList do
  if ((n.bInitial) && (n.iState & Node.MOVED == 0) && (n.iState & Node.RENAMED == 0)&& (n.iState & Node.DELETED == 0)) then
    iCounter++;
  end if
  if ((n.bInitial) && ((n.iState & Node.MOVED == Node.MOVED) || (n.iState & Node.RENAMED == Node.RENAMED) || (n.iState & Node.DELETED == Node.DELETED))) then
    iCounter++;
  end if
  if ((n.bInitial == false) && ((n.iState & Node.MOVED == Node.MOVED) || (n.iState & Node.RENAMED == Node.RENAMED) || (n.iState & Node.DELETED == Node.DELETED))) then
    iCounter++;
  end if
end for
```

Analogous, we can calculate the number of high-level changes. For this, we define a class *ProcessPart* with attributes that describe the state of a process model part (*bInitialPart* of type *boolean*, *DELETEDPart* of type *int* and *MOVEDPart* of type *int*). For instance, if *bInitialPart* is *false* then a new process model part has been inserted in the process model. Else no change has been performed.

To query the state of a process model part one can use the variable *iStatePart* of type *int*. The *list processpartList* is used to store all process model parts of a business process model. The algorithm for process model parts can be structured in the same way as in Algorithm 1. In the algorithm for calculating the number of high-level changes, case 1 would be valid, if a whole business process model has been reused by another user without any change. Case 2 would be valid, if the state of a process model part stored in the process repository has been modified by moving or deleting a whole process model part. In Case 3 the algorithm (to calculate the number of high-level changes) counts the number of newly inserted process model parts and the number of movements or deletions of whole process model part being initiate available in the process model.

Based on these two algorithms for primitive and high-level changes we can provide process builders with an alert function. The capability of this function is to inform users when specific changes have been performed. If a process builder subscribes to the alert function then she will receive reliable information about primitive, high-level changes and newly stored process models.

Figure 2a shows the interface of the alert function of the social networks enhanced recommendation system. The process builder can decide if she is interested in an alert on high-level or primitive changes or on newly inserted process model (parts) in the repository. The user can apply the alert function for all process models, respectively parts. Process models and model parts are listed

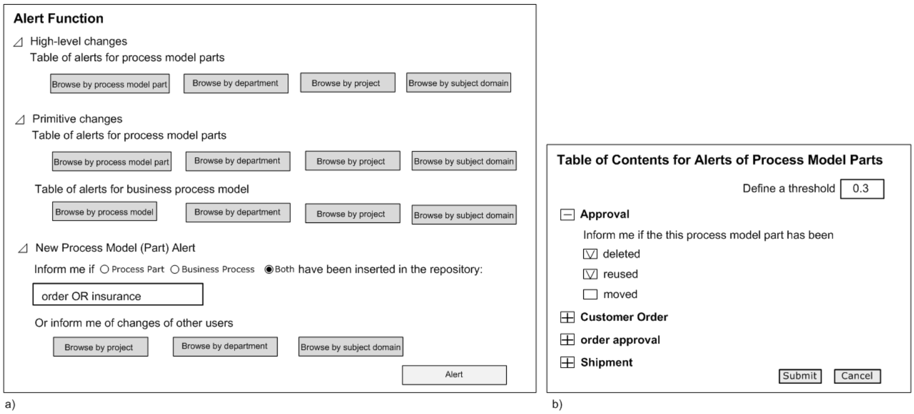


Fig. 2. Table of Contents of Pull Services

alphabetically, or in case that process builders are working in organizational boundaries they can be listed by departments, projects or subject domains.

The alert function on new process model (parts) can be activated when filling in keywords in a text field or when browsing by the project, the department or the subject domain. Then the system will automatically send a notice without a specific request from the subscriber if process builders with specific metadata have inserted a process model matching the user's keywords. In Figure 2a the user wants to be informed if process models and parts regarding *order* or *insurance* have been stored in the repository.

To determine relevant process models and parts we use the keyword respectively tag extraction method explained in [2]. Initially, we extract the words and remove common stop words, which yields the set t_{raw}. The remaining tag candidates t_{raw} are then expanded with their related synonym sets, which are determined via WordNet[2], resulting in the set t_{query}. The relevant process models and parts are then determined by querying an index, where the query term is the concatenation of all tags in the set t_{query}. Whenever new models are stored in the repository the system poses the query t_{query} and notifies the process builder if a new process model matching the process builder's criteria has been stored.

If process builders decide to be informed about primitive changes in process models then they have to specify a threshold, i.e., a value that indicates a degree of significant changes in a process model. In Figure 2b the user indicated a threshold of 0.3., that means that primitive changes should be considered if 30% of elements in the process model *Process for checking orders* have been deleted or renamed. A threshold of 0.3 tends to be coherent. If a process builder wants to be informed about high-level changes, a threshold like this will not be necessary but the rest of the interface will be similar.

[2] http://wordnet.princeton.edu/

4.2 Push Service

The push service of the recommendation system allows process builders to notify other process builders when changes have been performed or relevant content has been stored in the repository. Figure 3 shows the interface for the push service. The process builder can either push the information to her public social networks, to process builders belonging to a subject domain, a work environment or even to specific process builders (e.g., friends). We assume that the number of possible

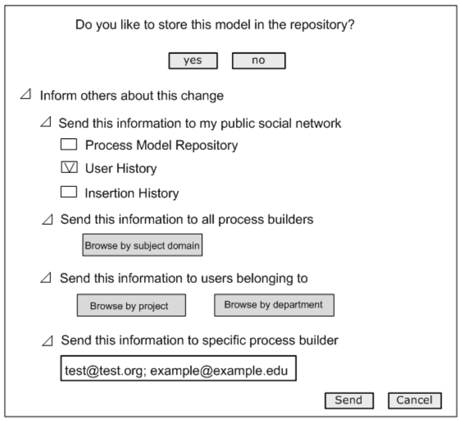


Fig. 3. Table of Contents of the Push Service

domains and work environments is usually fixed within a company and hence we can provide the process builder with an interface where she can choose from a list of subject domains and work environments to whom the information should be forwarded. Whenever the process builder stores her model in the repository she can activate the push service for this model.

5 Related Work

The initial idea of a recommendation-based process modeling support system has been described in [2]. An extension of the system is presented in [5], which considers the modeling perspective when suggesting appropriate recommendations. This paper extend the previous works in the perspective of social networks.

In the field of social network, privacy has been studied in several approaches [6,7]. [6] proposes a model of privacy for reconstructing a graph privately; the approach does focus more on security aspects and uses a completely different metamodel than the one presented in this paper. The approach in [7] could be used in our scenario in order to join two (related) personalized social networks. Such a join could be regarded as an extension of our approach, which would provide synergy effects for process builders. Regarding the searching of expertise in social networks, the set of approaches found in the literature [8,9] follow a different metamodel and use different kinds of social networks. E.g., [9] retrieves the expertise through out semi-structured content. [8] e.g., uses organization's email data set. The work in [10] is the first approach to apply social networks in the BPM area. However, their focus is not on process modeling, but on the analysis of interpersonal relationships in an organization.

Push service and *pull service* are related to *alert service* and *propagation*. Alert services are particularly known for digital libraries [11] where users want to be notified whenever a new publication in a specific research area is published. In our context, we use the alert service to notify process builders about changes in process models and thus we use a different basis for our algorithms. Propagation in networks have been suggested for innovations [12] or effects of a change in technology [13]. These approaches have a different focus than the one presented in this paper. Approaches presented for pull and push technologies on the web [14] or for mobile devices [15] can not be applied in our context.

6 Conclusion and Reflection

The contribution of this paper is that it presents new social features of a recommendation system for process builders. Specifically, user-specific information can be added to member profiles so that modelers can build personalized views on their social networks. Furthermore, facilities are described to propagate relevant changes through such social networks. Both types of facilities are expected to increase the 'sociability' of that recommendation system, such that process builders can more easily leverage the modeling efforts by their peers.

Currently, we are implementing the described features into our recommendation system and plan to initiate a new round of practical evaluations under realistic process modeling conditions. It is our belief, cf. [16], that only through integrating both design and evaluation activities it is possible to develop our ideas towards a truly supportive recommendation system for process builders.

There is notable increase of enterprises that possess large repositories of process models which they manage and extend as valuable assets, see e.g. [17]. While much of the current research on process modeling focuses on issues with a single process model, the demand will rise for insights on how to extend, maintain, and disclose such large sets of process models. We hope that our work will contribute to the development of systems that can deal with these needs.

References

1. Wasserman, S., Faust, K., Iacobucci, D., Granovetter, M.: Social Network Analysis: Methods and Applications. Cambridge University Press, Cambridge (1994)
2. Hornung, T., Koschmider, A., Lausen, G.: Recommendation based process modeling support: Method and user experience. In: Li, Q., Spaccapietra, S., Yu, E., Olivé, A. (eds.) ER 2008. LNCS, vol. 5231, pp. 265–278. Springer, Heidelberg (2008)

3. Koschmider, A., Song, M., Reijers, H.A.: Social Software for Modeling Business Processes. In: Ardagna, D., et al. (eds.) BPM 2008 Workshops. LNBIP, vol. 17, pp. 666–677. Springer, Heidelberg (2009)
4. Weber, B., Reichert, M., Rinderle-Ma, S.: Change patterns and change support features - enhancing flexibility in process-aware information systems. Data & Knowledge Engineering 66, 438–466 (2008)
5. Koschmider, A., Habryn, F., Gottschalk, F.: Real support for perspective-compliant business process design. In: Ardagna, D., et al. (eds.) BPM 2008 Workshops. LNBIP, vol. 17, pp. 32–43. Springer, Heidelberg (2009)
6. Frikken, K.B., Golle, P.: Private social network analysis: how to assemble pieces of a graph privately. In: Proceedings of the 5th ACM workshop on Privacy in electronic society, pp. 89–98. ACM, New York (2006)
7. Brickell, J., Shmatikov, V.: Privacy-preserving graph algorithms in the semi-honest model. In: Roy, B. (ed.) ASIACRYPT 2005. LNCS, vol. 3788, pp. 236–252. Springer, Heidelberg (2005)
8. Zhang, J., Ackerman, M.S.: Searching for expertise in social networks: a simulation of potential strategies. In: Proceedings of the 2005 international ACM SIGGROUP conference on Supporting group work, pp. 71–80. ACM, New York (2005)
9. Tsai, T.M., Shih, S., Stu, J., Wang, W.N., Chou, S.C.T.: Beyond web-log: Transform blog into personal expertise and social network via myfoaf support. In: Workshop on Social Web Search and Mining (2008)
10. van der Aalst, W.M.P., Reijers, H.A., Song, M.: Discovering social networks from event logs. Computer Supported Cooperative Work 14, 549–593 (2005)
11. Faensen, D., Faulstich, L., Schweppe, H., Hinze, A., Steidinger, A.: Hermes: A notification service for digital libraries. In: Proceedings of the 1st ACM/IEEE-CS joint conference on Digital libraries, pp. 373–380 (2001)
12. Mason, W.A., Jones, A., Goldstone, R.L.: Propagation of innovations in networked groups. Journal of Experimental Psychology 137, 233–422 (2008)
13. Burkhardt, M.E., Brass, D.J.: Changing patterns or patterns of change: The effect of a change in technology on social network structure and power. Administrative Science Quarterly 35, 104–127 (1990)
14. Kendall, J.E., Kendall, K.E.: Information delivery systems: an exploration of web pull and push technologies. Communications of the Association for Information Systems 1 (1999)
15. Qaddour, J.: Wap and push technology integrated into mobile commerce applications. In: Proceedings of the IEEE International Conference on Computer Systems and Applications, Washington, DC, USA, pp. 779–785. IEEE Computer Society, Los Alamitos (2006)
16. Hevner, A., March, S., Park, J., Ram, S.: Design Science in Information Systems Research. Management Information Systems Quarterly 28, 75–106 (2004)
17. Reijers, H., Mans, R., van der Toorn, R.: Improved Model Management with Aggregated Business Process Models. Data & Knowledge Engineering 68, 221–243 (2009)

Computer Support for Agile Human-to-Human Interactions with Social Protocols

Willy Picard

Poznań University of Economics,
Department of Information Technologies,
ul. Mansfelda 4, Poznań, Poland
Willy.Picard@ue.poznan.pl
http://www.kti.ue.poznan.pl/

Abstract. Despite many works in CSCW, groupware, workflow systems and social networks, computer support for human-to-human interactions is still insufficient, especially support for agility, i.e. the capability of a group of human beings, collaborators, to rapidly and cost efficiently adapt the way they interact to changes. In this paper, requirements for computer support for agile H2H interactions are presented. Next, the concept of social protocol is proposed as a novel model supporting agile H2H interactions. A social protocol consists of an extended social network and a workflow model.

Keywords: Social protocols, workflow, adaptation, social networks, human-to-human. interactions.

1 Introduction

Computer support for Human-to-Human (H2H) interactions has a long history in computer science: from early visionary ideas of Douglas Engelbart at the Stanford Research Institute's Augmentation Research Center on groupware in the 60's, through CSCW and workflows in the 80's, and with social network sites in the 2000's. However, computer support for agile H2H interactions is still insufficient in most collaborative situations.

Among various reasons for the weak support for H2H interactions, two reasons may be distinguished: first, many *social elements* are involved in the H2H interaction. An example of such a social element may be the roles played by humans during their interactions. Social elements are usually difficult to model, e.g. integrating hierarchical relations among collaborators to collaboration models. A second reason is the *adaptation capabilities* of humans which are not only far more advanced than adaptation capabilities of software entities, but also are not taken into account in existing models for collaboration processes.

Agility in H2H interactions refers to the capabilities of a group of human beings, collaborators, to rapidly and cost efficiently adapt the way they interact to changes. Changes may occur:

– within the group: e.g., a collaborator may be temporary unavailable or he/she may acquire new skills,

W. Abramowicz (Ed.): BIS 2009, LNBIP 21, pp. 121–132, 2009.

– in the environment of the group: e.g., a breakdown of a machine may occur, weather conditions may prevent the realization of a given task.

In this paper, we present a model which provides support for agile H2H interactions based on the concept of social protocols. In Section 2, requirements for a computer support for agile H2H interactions are presented. Next, the concept of social protocols supporting agile H2H interactions is detailed. Then, the proposed solution is discussed. Finally, Section 5 concludes the paper.

2 Requirements for Support for Agile H2H Interactions

2.1 A Model of the Social Environment

A first requirement for support for agile H2H interaction is the modeling of the *social environment* within which interactions take place. H2H interactions imply the involvement of at least two collaborators, each of them having her/his own social position. By social position, we mean a set of interdependencies with entities (generally individuals or organizations): e.g. a collaborator has a set of colleagues, works in a given company, belongs to a family.

Agility during H2H interactions implies a rapid adaptation of the collaboration group to new conditions. The social environment is a core tool in the adaptation process as it provides information about available resources collaborators are aware of:

– within the group: e.g., if a collaborator is temporary unavailable, another person in the social environment may substitute for the unavailable collaborator,
– in the environment of the group: e.g., if weather conditions prevent the realization of a given task, new collaborators which were not initially involved in the realization of the cancelled task may be needed to overcome it.

A partial answer to the question of modeling a social environment may be found in popular in the last five years social network sites, such as LinkedIn [17], MySpace [18], Orkut [19], Friendster [10] and Facebook [8], to name a few. Boyd and Ellison [6] define social network sites as “web-based services that allow individuals to (1) construct a public or semi-public profile within a bounded system, (2) articulate a list of other users with whom they share a connection, and (3) view and traverse their list of connections and those made by others within the system.” The second and third points of this definition illustrate a key feature of social network sites, i.e. social network sites allow users for an easy access to information about persons they know (friends, colleagues, family members) and potentially about contacts of these persons.

However, the model of social environment adopted in social network sites captures only interdependencies among individuals or organizations. The interdependencies with information systems, e.g. web services, are an important element of the landscape of H2H interactions: while individuals represent the “who” part of H2H interactions, information systems usually represent the “how” part. A collaborator (the *individual*) performs some activity with the help of a tool (the *information system*). Therefore, we claim that a model of the social environment for H2H interactions should integrate both interdependencies among collaborators and interdependencies among collaborators and information systems.

A model of social environment integrating interdependencies among collaborators and among collaborators and information systems would allow collaborators to react to new situations not only by changing the set of collaborators, but also by changing the set of tools. Additionally, such a model would allow collaborators for agility with respect to changes related with information systems: e.g., if an information system is unavailable, collaborators may seek for an alternative in their social environment.

2.2 Structured H2H Interactions

Supporting agile H2H interactions requires guidance for collaborators about tasks they may perform at a given moment of time. Such a guidance allows collaborators for *focusing on appropriate tasks* that need to be fulfilled at a given moment of time, in a given collaboration situation, instead of facing all potential tasks that they may perform.

The tasks that a given collaborator may perform depend also on the *role* he/she is playing within a given group. Therefore support for agile H2H interactions implies the mapping between collaborators and roles they are playing within a given group.

Additionally, H2H interactions are often structured according to collaborative patterns [3,23]. In similar situations, in different groups, collaborators perform activities whom successiveness is identical among the various groups: e.g., a brainstorming session consists usually of five phases:

1. the chairman presents the problem,
2. every participant presents his/her ideas,
3. the chairman classifies the ideas,
4. every participant may comment any idea,
5. the chairman summarizes the brainstorming session.

In the former example, each phase may be decomposed as a sequence of activities to be performed, with activities associated to roles. H2H interactions could therefore be structured with the help of a *process* and an associated *process model* specifying the sequences of activities, the association between activities and roles, and the mapping between collaborators and roles.

Results of studies in workflow technology and process modeling [9,15,11,5] provide a strong foundation for support for structured H2H interactions based on the concepts of workflow and process models.

2.3 Layered Interaction Models

The concept of process model presented in the former subsection as a mean to structure H2H interaction has to be considered at three levels of abstraction:

- *abstract process model*: a process model is abstract if it defines the sequence of activities to be potentially performed by collaborators playing a given role, without specifying neither the implementation of activities, nor the attribution of roles to collaborators. As an example, an abstract process model for a brainstorming session may specify that, first, a chairman presents the brainstorming session problem, next, participants present their ideas. Neither the implementation of the presentation of the problem and participants' ideas, nor the group collaborators are defined in the abstract process model.
- *implemented process model*: a process model is implemented if it defines the implementation of activities defined in an associated abstract process model. As an example, an implemented process model based on the brainstorming abstract process model formerly presented may specify that the presentation of the brainstorming session problem will be implemented as the sending of an email to all participants, while the presentation of ideas will be performed as posts to a forum.
- *instantiated process model*: a process model is instantiated if the attribution of roles to collaborators for a given implemented process model has been set. Additionally, an instantiated process model, referred also as *process instance*, keeps trace of the current state of the H2H interactions. As an example, the former implemented process model may be instantiated by specifying who plays the chairman role and who are the participants. Additionally, the process retains its current state which may for instance be "participants are presenting ideas".

The following analogy with object-oriented programming illustrates the three levels of abstraction presented above:

- abstract process models are similar to interfaces or abstract classes. An abstract process model does not rely, nor provide an implementation of activities, as an interface does not provide an implementation of methods;
- implemented process models are similar to classes. An implemented process model provides an implementation of activities, as a class provides an implementation of methods.
- instantiated process models are similar to objects. An instantiated process model rules the H2H interactions according to a given implemented process model and has its own state, as an object behaves according to its class and has its own state too.

The separation of these three levels of abstraction leads to *process model reuse*. By separating the logical structure of H2H interactions from its implementation, an abstract process model may be reuse in various contexts, IT environments, groups of collaborators. As a consequence, a group of collaborators facing some unpredicted situation may identify an already defined abstract or implemented process model allowing them to solve their problem. Then, the group may react rapidly by just (eventually implementing and) instantiating the process. The brainstorming process presented above is an example of an abstract or implemented process that may be reuse by various groups of collaborators to interact in an agile way.

2.4 Adaptability

Adaptability is a core requirement of support for agile H2H interactions. Adaptability refers in this paper to the capability of a group of collaborators to modify *at run-time* the process model ruling their interactions.

In typical workflow management systems, two parts may be distinguished: a *design time* part allows for definition of workflow schemas while the *run-time* part is responsible for execution of workflow instances. A main limitation of typical workflow management systems is the fact that once a workflow schema has been instantiated, the execution of the workflow instance must stick to the workflow schema till the end of the workflow instance execution. This limitation is not an issue if the lifespan of workflow instances is short in comparison with the time interval between two requests for changes of the workflow schema. When the lifespan of workflow instances is long in comparison with the time interval between two requests for changes of the workflow schema, a high number of workflow instances has to be executed with an "incorrect" workflow schema (i.e. that does not take into account required changes) or cancelled. As a consequence, typical workflow management systems are not flexible enough to support collaborative processes in two cases: highly dynamic, competitive markets/environments and long lasting collaboration processes.

In the case of highly dynamic, competitive markets/environments or long lasting collaboration processes, there is a strong need for the possibility to modify a workflow instance at run-time. Such modifications are usually needed to deal with situations which have not been foreseen nor modeled in the associated workflow schema. Adaptation refers to the possibility to modify a running instantiated process model to new situations which have not been foreseen and modeled in the associated abstract/implemented process model.

3 Social Protocols

Computer support for agile H2H interactions requires novel models to support requirements presented in Section 2. The solution presented in this paper is based on the concept of *social protocol*. This concept has been presented first in 2006 [22], based on the concept of *collaboration protocol* [20]. An extended version of the concept of social protocol, including elements related with the modeling of the social environment, is presented in this paper.

3.1 Abstract Social Protocols

An abstract social protocol, SP_a, consists of two parts:

- an *abstract social network*: a direct graph modeling interdependencies among *abstract resources*. An abstract social network models the social environment required for a particular collaboration pattern.
- an *abstract interaction protocol*: a direct graph modeling interdependencies among *abstract activities*. An abstract interaction protocol models the sequence of activities in a particular collaboration pattern.

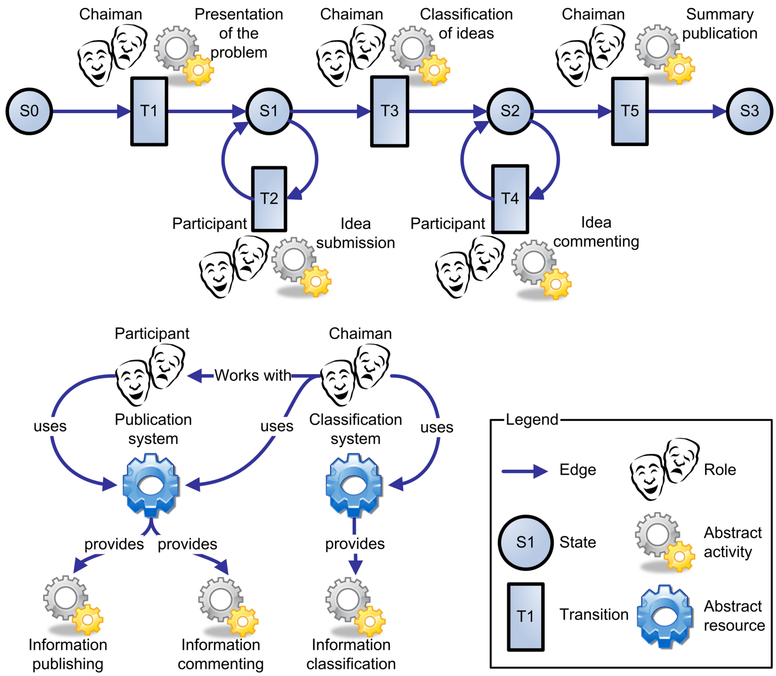


Fig. 1. An example of an abstract social protocol. At the top, the abstract interaction protocol of a brainstorming session. At the bottom, the abstract social network.

In an abstract social network, vertices represent abstract resources that may support or be actively involved in the collaboration process, such as a collaboration role or a class of information systems. Edges represents relations between resources associated with social interaction types, such as "works with", "has already collaborated with" among roles, or "is the owner", "uses" between a role and a class of information systems. Labels associated with edges are not predefined, as the concept of social protocol should be flexible enough to encompass new types of interdependencies among resources. Therefore, new labels may be freely created at design time.

An example of an abstract social protocol for brainstorming sessions is presented in Figure 1.

Formally, an abstract social network, SN_a, is a directed graph $< R_a, SI_a >$ where R_a is a finite set of nodes, each node referring to an abstract resource, SI_a the social interdependencies relation $SI_a \subseteq R_a \times R_a \times SIT$, with SIT: a set of social interaction types.

In an abstract interaction protocol, vertices represents:

– abstract activities that may be performed during the collaboration process, such as "present the brainstorming problem" or "present an idea". Activities are associated with a given role, e.g. only the chairman may present the brainstorming problem;

– states in which the group may be at various moments of the collaboration process, e.g. the group may be “waiting for ideas”.

Edges run between activities and states, never between activities nor between states. Edges capture the potential activities in a given state, or states after the execution of a given activity. One may recognize in abstract interaction protocols the concept of Petri nets, where states are places and activities/roles pairs are transitions.

Formally, an abstract interaction protocol, IP_a, is a 3-tuple (T_a, S_a, E_a), where T_a is a finite set of abstract transitions, S_a is a finite set of states, and E_a is a multiset of arcs $E_a : (T_a \times S_a) \cup (S_a \times T_a) \to \mathbb{N}$. An abstract transition $t_a \in T_a$ consists of an abstract activity $a_a \in A_a$ and a role $\rho_a \in Roles_a$.

3.2 Implemented Social Protocols

Similarly to the relation between implemented process models and abstract process models presented in Section 2.3, an implemented social protocol defines the implementation of abstract activities associated with an abstract social protocol.

Therefore, an implemented social protocol consists of three parts:

– an abstract social protocol,
– a mapping of *abstract resources* associated to with abstract activities to *implemented resources*. For instance, the abstract resource “Publication system” of the former example may be mapped to a forum system on a given server.
– a mapping of *abstract activities* to *implemented activities*. For instance, the abstract activity “presentation of the problem” of the former example may be mapped to the URL of the form used to post information on the formerly mentioned forum system.

These two mappings may be built based on a pre-existing social environment defining interdependencies among resources (abstract and implemented). Additionally, the pre-existing social environment may be extended by the addition of missing resources. Therefore, on the one hand, the implementation procedure may take advantage of the social environment, on the other hand, the social network may benefit from the implementation procedure.

Formally, an implemented social protocol, SP_i, is a 5-tuple $(SP_a, R_i, A_i, \mathcal{R}_\downarrow, \mathcal{A}_\downarrow)$, where SP_a is an abstract social protocol, R_i is a finite set of implemented resources, A_i is a finite set of implemented activities, $\mathcal{R}_\downarrow : R_a \to R_i$ is a mapping function, such that $\forall r_a \in R_a, \exists r_i \in R_i, \mathcal{R}_\downarrow(r_a) = r_i$, and $\mathcal{A}_\downarrow : A_a \to A_i$ is a mapping function, such that $\forall a_a \in A_a, \exists a_i \in A_i, \mathcal{A}_\downarrow(a_a) = a_i$.

3.3 Social Processes

Similarly to the relation between instantiated process models and implemented process models presented in Section 2.3, a social process defines the implementation of abstract roles associated with an implemented social protocol, as well as keeps trace of the state of the H2H interactions.

Therefore, a social process consists of three parts:

- an implemented social protocol,
- a mapping of *abstract resources* associated with roles to *collaborators*. For instance, the abstract resource "brainstorming chairman" is mapped to collaborator "John Smith".
- a *marking* of active states.

The role-collaborator mapping may be built based on the pre-existing social environment. Additionally, the pre-existing social environment may be extended by the addition of missing resources, by the addition of collaborators. Therefore, on the one hand, the instantiation procedure may take advantage of the social environment, on the other hand, the social network may benefit from the instantiation procedure.

Formally, a social process, π, is a -tuple $(SP_i, C, \mathcal{C}_{\downarrow}, M)$, where SP_i is an implemented social protocol, C is a finite set of collaborators, $\mathcal{C}_{\downarrow} : Roles_a \rightarrow C$ is a mapping function, such that $\forall \rho_a \in Roles_a, \exists c \in C, c \in \mathcal{C}_{\downarrow}(\rho)$, and $M : S_a \rightarrow \mathbb{N}$ is a marking assigning a number of tokens to each state.

The creation of a social process requires: 1) the choice of an implemented social protocol, 2) the attribution of roles to collaborators, and 3) the creation of an *initial marking*. The initial marking contains a set of active states, referred as initial states.

Next, the social process rules the interactions according to the associated social protocol:

- a collaborator may trigger a transition t if, 1) he/she plays the associated role, 2) all states from which at least one arc leads to t contain a token, i.e. the marking contains all states from which at least one arc leads to t.
- when a collaborator triggers a transition, 1) he/she performs the associated activity, 2) the marking is updated, i.e. the tokens from input states (those states from which at least one arc leads to t) are removed, and new tokens are created in output states (those states to which at least one arc comes from t).

3.4 Meta-processes

The concept of *meta-process* is our answer to the adaptation requirement. During the execution of an instantiated social protocol, collaborators may identify a need for modification of the process instance they are involved in. As a consequence, collaborators need to interact to decide how the process should be changed. A meta-process is a social process associated with another social process π allowing collaborators of π to decide in a structured collaborative way how the process π should be modified.

Formally, a meta-process μ is a pair π, π^{α}, where both π, the to-be-adapted process, and π^{α}, the adaptation process, are social processes, share the same set of collaborators C. The social process π^{α} rules H2H interactions concerning changes to be performed in π.

Table 1. The effects of various change types on the social process, the implemented social protocol, and the abstract social protocol, and the potential need for a redefinition of activities implementation

Level	Type	Instance	Implemented	Abstract	Implementation
1	role attributions	X			
2	activity implementation	X	X		X
3	structural simplification	X	X	X	
4	structural modification	X	X	X	
5	structural extension	X	X	X	X

Depending on the type of changes provided during the meta-process, five situations (summarized in Table 1) may be distinguished:

– Level 1 changes – *role attributions*: a meta-process may result in changes in the role-collaborator mapping. For instance, "Susan Doe" may replace "John Smith" as the brainstorming chairman. Such changes have an influence only on the instantiated process model;
– Level 2 changes – *activity implementation*: a meta-process may result in changes in the mapping of abstract activities to implemented activities. For instance, instead of publishing new ideas on a given forum, collaborators may decide to publish their ideas using a mailing list at a given address. Such changes imply not only modifications of the instantiated process model, but also modifications of the associated implemented social model;
– Level 3 changes – *structural simplification*: a meta-process may result in changes towards the simplification of the structure of the interaction protocol, with activities, states, transitions, roles or edges removed from the interaction protocol. For instance, collaborators may decide that the summary of the brainstorming session is not required. Such changes imply modifications of the instantiated, the associated implemented, and the abstract social protocol. However, no new implementation has to be provided.
– Level 4 changes – *structural modification*: a meta-process may result in changes towards the modification of the structure of the interaction protocol such that no new activity, state, or role have to be defined. Edges and transitions can be freely modified to reflect a new organization of states, activities and roles. For instance, collaborators may decide that any participant may summarize brainstorming session, and not only the chairman. Such changes imply here also modifications of the instantiated, the associated implemented, and the abstract social protocol. However, no new implementation has to be provided.
– Level 5 changes – *structural extension*: a meta-process may result in changes towards the extension of the structure of the interaction protocol by the addition of new activities, states, or roles. Edges and transitions can be freely modified to reflect a new organization of states, activities and roles. For instance, collaborators may decide that the classification of the ideas should be accepted by an "observer". This change implies the creation and the attribution of the role of "observer", the creation of two new activities

"accept classification" and "reject classification", as well as the choice of the implementation of these activities, and finally the creation of appropriate transitions. Such changes imply here also modifications of the instantiated, the associated implemented, and the abstract social protocol. Implementation of newly added activities has to be provided, and the role-collaborator mapping has to be redefined for newly created roles.

4 Discussion

Some interesting works have been done in the field of electronic negotiations to model electronic negotiations with the help of negotiation protocols. In [16], it is stated in that, in the field of electronic negotiations, "the protocol is a formal model, often represented by a set of rules, which govern software processing, decision-making and communication tasks, and imposes restrictions on activities through the specification of permissible inputs and actions". One may notice the similarity with the concept of social protocol. The reason for this fact is that the model presented in this paper was originally coming from a work on protocols for electronic negotiations [21]. However, to our knowledge, none of the works concerning negotiation protocols provides support for the modeling of the social environment. Moreover, these works are by nature limited to the field of electronic negotiations which is just a subset of the field of H2H interactions.

As process modeling is concerned, many works have already been conducted in the research field of workflow modeling and workflow management systems. Many works [1,2,25,12,13] have focused on formal models and conditions under which a modification of an existing – and potentially running – workflow retains workflow validity, the ADEPT2 project[7] being probably the most advanced one. However, to our best knowledge, current works concerning workflow adaptation focus on interactions, and the importance of social aspects are not or insufficiently taken into account by these works.

Sadiq and al.[24] have proposed an interesting model for flexible workflows, where flexibility refers to "the ability of the workflow process to execute on the basis of a loosely, or partially specified model, where the full specification of the model is made at runtime, and may be unique to each instance." However, support for flexibility does not ensure support for adaptability, as flexibility, as proposed by Sadiq and al., implies that the workflow designer has specified at design time frames and boundaries to possible modifications of the workflow.

5 Conclusion

While many works are currently done on modeling collaboration processes in which software entities (agents, web services) are involved, modeling collaboration processes in which mainly humans are involved is an area that still requires much attention from the research community. Some of the main issues to be addressed are the social aspects of collaboration and the adaptation capabilities of humans. In this paper, the requirements of computer support for agile H2H interactions are presented. Additionally, the concept of social protocol, combining social networks and workflow models, is proposed as a model supporting agile H2H interactions.

The main innovations presented in this paper are 1) the requirements for agile H2H interactions, 2) the refinement of the concept of social protocol by the addition of the concept of social network as a way to model the social environment, and 3) the three-layer view on social protocols – abstract, implemented, and instantiated – and the concept of meta-process.

A prototype, based on Dyng [14], is currently under implementation to validate the model presented in this paper. Among future works, methods to update the social network to reflect H2H interactions performed in a given process are still to be proposed.

References

1. van der Aalst, W.M.P.: The Application of Petri Nets to Workflow Management. J. of Circuits, Systems and Computers 8(1), 21–66 (1998)
2. van der Aalst, W.M.P., Basten, T., Verbeek, H.M.W., Verkoulen, P.A.C., Voorhoeve, M.: Adaptive Workflow: On the Interplay between Flexibility and Support. In: Filipe, J. (ed.) Proc. of the 1st International Conference on Enterprise Information Systems, vol. 2, pp. 353–360. Kluwer Academic Publishers, Dordrecht (1999)
3. van der Aalst, W.M.P., van Hee, K.M., van der Toorn, R.A.: Component-Based Software Architectures: A Framework Based on Inheritance of Behavior. BETA Working Paper Series, WP 45, Eindhoven University of Technology, Eindhoven (2000), http://wwwis.win.tue.nl/~wvdaalst/publications/p108.pdf
4. van der Aalst, W.M.P., Weske, M., Wirtz, G.: Advanced Topics in Workflow Management: Issues, Requirements, and Solutions. J. of Integrated Design and Process Science 7(3), 47–77 (2003)
5. van der Aalst, W.M.P., van Hee, K.: Workflow Management: Models, Methods, and Systems (Cooperative Information Systems). MIT Press, Cambridge (2004)
6. Boyd, D.M., Ellison, N.B.: Social Network Sites: Definition, History, and Scholarship. J. of Computer-Mediated Communication 13(1), 210–230 (2007)
7. Dadam, P., Reichert, M.: The ADEPT Project: A Decade of Research and Development for Robust and Flexible Process Support. Technical Report. Fakultät für Ingenieurwissenschaften und Informatik, Ulm (2009), http://dbis.eprints.uni-ulm.de/487/1/Reichert_01_09-2.pdf
8. Facebook, http://www.facebook.com/
9. Fisher, L.: 2007 BPM & Workflow Handbook. Future Strategies Inc. (2007)
10. Friendster, http://www.friendster.com/
11. Harrison-Broninski, K.: Human Interactions: The Heart And Soul Of Business Process Management: How People Really Work And How They Can Be Helped To Work Better. Meghan-Kiffer Press (2005)
12. ter Hofstede, A.H.M., Orlowska, M.E., Rajapakse, J.: Verification Problems in Conceptual Workflow Specifications. Data Knowledge Engineering 24(3), 239–256 (1998)
13. ter Hofstede, A.H.M., Orlowska, M.E., Rajapakse, J.: Verification Problems in Conceptual Workflow Specifications. In: Thalheim, B. (ed.) ER 1996. LNCS, vol. 1157, pp. 73–88. Springer, Heidelberg (1996)

14. Huriaux, T., Picard, W.: DynG: a Multi-protocol Collaborative System. In: Funabashi, M., Grzech, A. (eds.) Proc. of the 5th IFIP International Conference on e-Commerce, e-Business, and e-Government (I3E 2005), pp. 591–605. Springer, Heidelberg (2005)
15. Jeston, J., Nelis, J.: Business Process Management, 2nd edn. Practical Guidelines to Successful Implementations. Butterworth-Heinemann (2008)
16. Kersten, G.E., Strecker, S.E., Lawi, K.P.: Protocols for Electronic Negotiation Systems: Theoretical Foundations and Design Issue. In: Proc. of the 5th Conference on Electronic Commerce and Web Technologies (ECWeb 2004), pp. 106–115. IEEE Computer Society, Los Alamitos (2004)
17. LinkedIn, http://www.linkedin.com/
18. MySpace, http://www.myspace.com/
19. Orkut, http://www.orkut.com/
20. Picard, W.: Modeling Structured Non-monolithic Collaboration Processes. In: Camarinha-Matos, L., Afsarmanesh, H., Ortiz, L. (eds.) Collaborative Networks and their Breeding Environments, the 6th IFIP Working Conference on Virtual Enterprises PRO-VE 2005, pp. 379–386. Springer, Heidelberg (2005)
21. Picard, W., Huriaux, T.: DynG: A Protocol-based Prototype for Non-monolithic Electronic Collaboration. In: CSCW in Design 2005. LNCS, vol. 3865, pp. 41–50. Springer, Heidelberg (2006)
22. Picard, W.: Computer Support for Adaptive Human Collaboration with Negotiable Social Protocols. In: Abramowicz, W., Mayr, H.C. (eds.) 9th International Conference on Business Information Systems BIS 2006. Lecture Notes in Informatics, vol. P-85, pp. 90–101. Gesellschaft für Informatik (2006)
23. Russell, N., ter Hofstede, A.H.M., Edmond, D., van der Aalst, W.M.P.: Workflow Resource Patterns. BETA Working Paper Series, WP 127, Eindhoven University of Technology, Eindhoven (2004), http://www.workflowpatterns.com/documentation/documents/Resource
24. Sadiq, S.W., Orlowska, M.E., Sadiq, W.: Specification and validation of process constraints for flexible workflows. Information Systems 30(5), 349–378 (2005)
25. Sadiq, S.W., Orlowska, M.E.: Analyzing process models using graph reduction techniques. Information Systems 25(2), 117–134 (2000)

A Requirements Engineering Approach for Data Modelling of Process-Aware Information Systems*

Jose Luis de la Vara[1], Michel H. Fortuna[2,3], Juan Sánchez[1], Cláudia M.L. Werner[3], and Marcos R.S. Borges[4]

[1] Centro de Investigación en Métodos de Producción de Software, Universidad Politécnica de Valencia, Spain
{jdelavara,jsanchez}@pros.upv.es
[2] Universidade Federal de Juiz de Fora, Brazil
michel.fortuna@ufjf.edu.br
[3] COPPE/Programa de Engenharia de Sistemas e Computação, Universidade Federal do Rio de Janeiro, Brazil
werner@cos.ufrj.br
[4] IM-NCE/Programa de Pós-Graduação em Informática
Universidade Federal do Rio de Janeiro, Brazil
mborges@nce.ufrj.br

Abstract. Business process modelling is a common activity when modelling an organization during the requirements engineering process of an information system. It is a must for modelling of process-aware information systems, and it can be considered the main activity. Nonetheless, business process diagrams must be complemented with a data perspective of these systems. This paper presents a requirements engineering approach that provides methodological guidance to meet this need through the integration of two other approaches. First, functional requirements are elicited and specified from business process diagrams. Next, the information flows of these requirements are specified. Finally, data modelling is carried out by following a set of guidelines.

Keywords: Process-aware information system, business process modelling, functional requirement, data modelling, info cases, BPMN.

1 Introduction

Understanding of the application domain is essential for the requirements engineering (RE) process of an information system (IS) for an organization. As a result, the need of organizational modelling has been widely acknowledged (e.g. [2][4][17]). Business process modelling is part of most of the organizational modelling-based approaches, and it is a must for the development of a process-aware information system (PAIS). PAISs manage and execute operational processes involving people, applications,

* This work has been developed with the support of the Spanish Government under the project SESAMO TIN2007-62894 and the program FPU AP2006-02324, and co-financed by FEDER. Marcos Borges was partially sponsored by a grant from CNPq # 304252/2008-5.

W. Abramowicz (Ed.): BIS 2009, LNBIP 21, pp. 133–144, 2009.

and/or information sources on the basis of process models [7], and their characteristics imply that RE approaches for their development must differ from traditional ones [1]. First, detailed process models are necessary in the RE process. Second, new systems must support new ways of (better) running an organization.

Although business process modelling can be considered to be the main activity for PAIS modelling, it is not sufficient to completely model a system. The behavioural perspective of a PAIS that is provided by business processes diagrams (BPD) must be complemented with a data perspective. Both functional requirements (which indicate what the system shall do and can be specified from business processes) and data models (which indicate what the system shall store) must be taken into account.

Functional requirements and data models provide different system perspectives; thus, they complement each other [19]. However, problems may arise when modelling both parts of a system. System analysts may have difficulties modelling data from functional requirements [20], and inconsistencies and contradictions may appear if data models and functional requirements are not properly managed [9].

This paper presents a RE approach (referred to as new approach hereafter) that provides methodological guidance for data modelling of a PAIS through the integration of two other approaches. The first approach is based on business process modelling and system purpose analysis [4][5] (referred to as business process-based approach hereafter), and it focuses on organizational modelling for the elicitation of functional requirements. The second approach, called info cases [8], focuses on the derivation of data models from functional requirements. As is detailed below, both approaches have been modified to make their integration possible.

The main purpose of the new approach is to help system analysts model the data of a PAIS from its functional requirements, which are elicited from the business processes of an organization. The specification of functional requirements and data modelling are integrated so that the problems described above do not arise.

On the one hand, the business process-based approach is well-suited for organizational modelling and elicitation of functional requirements, but it must be extended to address data modelling. On the other hand, the info cases approach is useful for data modelling from functional requirements, but it needs detailed guidance to elicit these requirements from BPDs. As a result, the new approach takes advantage of the features of both approaches and mitigates the weaknesses of their separate use.

The business process-based approach is the result of a collaborative project with the company CARE Technologies (http://www.care-t.com). CARE uses OO-Method [14], which is a methodology for automatic software generation based on data-centred conceptual modelling. Therefore, extending the approach with data modelling is essential in order to properly integrate it in the development process of the company, and the info cases approach has been chosen to meet this need.

The new approach starts from the business processes that an organization wants to execute. The BPDs are analysed to elicit and specify functional requirements of a PAIS, and then the information flows of the system for their execution are specified. Finally, the data of the PAIS are modelled in a domain class diagram (DCD), which is derived from the information flows by following a set of guidelines.

The rest of the paper is organized as follows: section 2 presents background and related work; section 3 describes the new approach; finally, section 4 explains our conclusions and future work.

2 Background and Related Work

This section describes the two RE approaches that have been integrated and presents related work to the new approach.

2.1 The Business Process-Based Approach

The new approach addresses organizational modelling by means of a RE approach [4][5] whose purpose is to avoid three problems of IS development for an organization: lack of understanding of the business and lack of focus on the purpose of the system by system analysts, and miscommunication with stakeholders.

The main characteristics of the approach are: 1) joint use of business process modelling and system purpose analysis; and 2) active stakeholders' participation. The approach consists of three stages, and it is based on the existence of a need or problem in the current organizational environment that could be fulfilled by an IS. The organization will change to fulfil the need, and business processes will be affected.

In the first stage, the current state of the organization for which an IS is going to be developed is modelled by means of a glossary, the business events, a domain data model (in the form of a class diagram in which just domain entities and relationships among them are modelled), the business rules, a role model, and a process map. BPDs are modelled from this information, and stakeholders must validate them.

The organizational need is analyzed during the purpose analysis stage. The aim is to find ways of fulfilling the need by means of the development or modification of an IS, and agree on the effect that the IS may have on the business processes with stakeholders. As a result, to-be BPD elements are labelled according to the IS support that they have, and changes in the business processes may occur. The labels of the BPD elements are: "O", if the element will not be part of the system; "IS", if the element will be controlled by the system; or "U", if the element will be controlled by a user.

Finally, functional requirements are specified by means of task descriptions, which detail the system support to business process tasks in a textual template.

2.2 The Info Cases Approach

In the new approach, data modelling is based on the info cases approach [8]. It provides an integrated model whose purpose is to jointly model use cases and domain (data) models in a single conceptual framework.

The approach has two main principles: 1) adoption of a special abstraction level called Informational Level of Objectives (ILO) to which a use case must belong to; and 2) systematic capture and precise description of the information flows between an IS and its actors. Use cases that conform to these principles are called info cases.

A use case is at the ILO if its realization allows a stakeholder to achieve a goal, which means causing a change of state in the system and/or its environment. When the goal is achieved, the state of the system must be steady, so no rollback to a previous state is necessary even if no other use case is subsequently activated.

The information flows of info cases are specified in a formalism that is capable of capturing the elements of a domain model and of permitting the identification of these elements. This formalism has two parts: a specification of the composition of flows, and a dictionary of elementary items of information.

As a result of belonging to the ILO and of precisely specifying the information flows, info cases capture the elements of a domain model and provide a set of semi-automatic rules for deriving it. According to their proponents, info cases can also increase the uniformity of the domain models produced by different modellers.

2.3 Related Work

The most common practice in the RE approaches that deal with functional requirements and data models is to model classes from uses cases or jointly with them. They are based on mechanisms such as linguistic patterns [6], sequence diagrams [10], activity graphs [11] or consistency guidelines [9]. These approaches are solution-oriented, so they do not properly analyse the application domain, nor do they provide guidance for the elicitation of functional requirements. When compared with the info cases approach, these approaches are more complex because they require the use of more models or models that are less flexible than information flows to obtain data models. In addition, they do not provide a homogeneous abstraction level for functional requirements such as ILO.

All the organizational modelling-based RE approaches that model business processes deal with functional requirements and data models (e.g. EKD [2] and ARIS [17]). However, they do not provide a homogeneous abstraction level for functional requirements, and they lack precise guidance for elicitation and specification of functional requirements and for assurance of consistency and completeness in data models. Some approaches focus on data modelling from BPDs (e.g. [16]), but the models that are obtained are incomplete and guidance for completion is not provided. When compared with the business process-based approach, these approaches do not explain how to improve business processes (which is essential for a PAIS), and they do not focus on system purpose or on communication with stakeholders.

Finally, several works have acknowledged the importance and benefits of a data-centred perspective when modelling business processes. They address issues such as the notion of business artifact [12], product-based workflow design of manufacturing processes [15], detection of data flow anomalies [18], and document-driven workflow systems [21]. The main difference of these approaches with the new one is that they take data into account from a perspective of data flow through tasks rather than from a perspective of information flows between a system and its actors. Also, they do not regard BPDs as a means for understanding the application domain and for the elicitation and specification of functional requirements.

3 Description of the New Approach

This section describes the new approach and explains how the business process-based approach and the info cases approach have been integrated.

Both approaches have been modified to make the integration possible. With regard to the business process-based approach, the BPMN notation [13] and the content of the textual template of task descriptions have been extended, and the granularity of the task descriptions is homogeneous as a result of adopting the ILO. With regard to the info cases approach, the information flows are created on the basis of the domain entities that are used in the task descriptions and the content of their textual templates, and the way of specifying information flows and the rules for the derivation of data models have been adapted.

The new approach (Fig. 1) consists of two stages: elicitation and specification of functional requirements, and data modelling. The first one is divided into modelling of the consecutive flows of to-be BPDs and specification of the task descriptions of a PAIS, whereas the second stage is divided into specification of the information flows of the task descriptions and modelling of the domain class diagram (DCD).

The new approach has been used and initially evaluated in several small/medium-size projects with CARE Technologies. As a case study, an actual rent-a-car company (http://www.rentacar-denia.com) is used. Nonetheless, the complete case study is not explained and just some of the information is used for the description of the new approach. The company is located in a tourist area, and its fleet of cars varies between the summer and the winter seasons. As a result, cars are usually bought at the beginning of the season and sold at the end. Its main activity is car rental, but it involves other activities (car maintenance, extras rental…).

The stages of the new approach and their activities are explained in the following subsections.

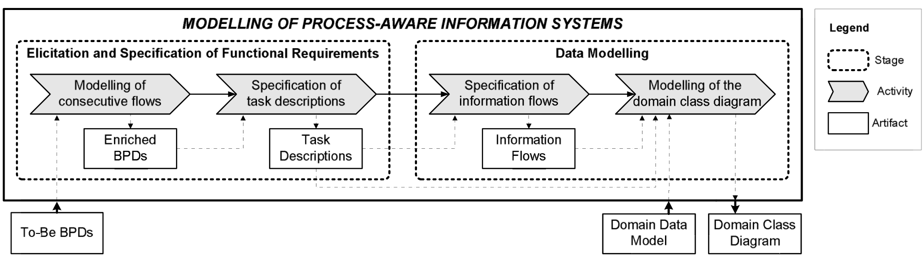


Fig. 1. RE approach for modelling of process-aware information systems

3.1 Elicitation and Specification of Functional Requirements

In the first stage of the new approach, it is necessary to elicit and specify the functional requirements of a PAIS from the business processes that an organization wants to execute according to its needs. Furthermore, these requirements must be part of the ILO so that the two RE approaches are integrated.

Before this stage is developed, the to-be business processes of an organization are modelled from the current business processes and the system purpose, and stakeholders collaborate in the process, as described in [5]. BPDs are completed with the textual specification of the business rules that have not been graphically modelled and a table that specifies the input and output data objects (domain entities in BPMN terminology) of the BPD tasks. All business rules and data objects are not always represented graphically to facilitate the understanding of the BPDs. Finally, BPD elements are labelled and system analysts and stakeholders agree upon the business rules and data objects that will be part of the PAIS. Afterwards, consecutive flows of to-be BPDs are modelled and task descriptions are elicited and specified.

3.1.1 Modelling of Consecutive Flows

To-be BPDs are analysed and enriched graphically by specifying the sequences of flow objects that are executed one after another without an interruption until the business processes have reached a steady state. This activity is necessary to elicit functional requirements that are part of the ILO.

These sequences of flow objects are modelled using a connecting object that is called consecutive flow, which does not exist in BPMN. The aim of this new type of connection is to graphically represent the fact that two flow objects are always executed consecutively. Its graphical representation is an arrow with two arrowheads.

The identification of consecutive flow is carried out as follows. For each sequence flow of a BPD, system analysts have to determine if the target flow object is always executed immediately after the source flow object when a token is in the sequence flow so that the business process reaches a steady state or advances in that direction. If so, both flow objects are linked by means of a consecutive flow.

Stakeholders' participation is essential to develop this activity. Stakeholders are the source of information from which the execution order of the flow objects is modelled, and they must validate that the consecutive flow has been properly modelled according to how the organization executes or wants to execute its business processes.

Fig. 2 shows an example of enriched BPD for the business process "car rental" of the rent-a-car company. Since the example is straightforward, the business process is not explained in great detail.

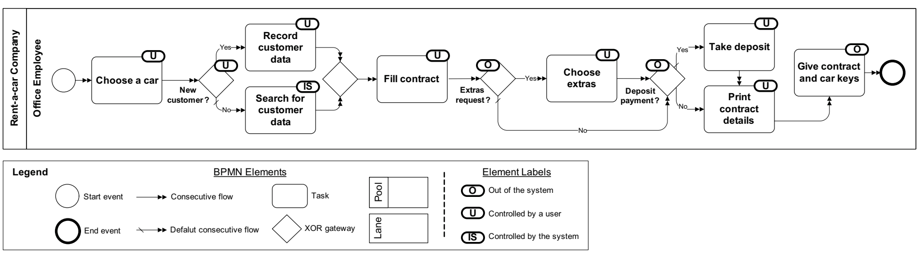


Fig. 2. Example of enriched business process diagram

3.1.2 Specification of Task Descriptions

In the second activity of the first stage, functional requirements are elicited from the enriched BPDs and specified in the form of task descriptions. Three issues must be pointed out. First, functional requirements are called task descriptions because their purpose is to specify adequate support for business process tasks, whereas the purpose of use cases is to specify required usages of a software system or interactions with it. Second, task descriptions must belong to the ILO, and this condition is assured thanks to the analysis of consecutive flows. Third, unlike the business process-based approach, the granularity of BPD tasks and task descriptions may not be the same. A task description can support several BPD tasks.

A task description specifies the support of a PAIS to the execution of a set of consecutive flow objects that denote tasks. These flow objects will be controlled by the same user role or by the system, and they jointly represent a functional requirement that belongs to the ILO. As a consequence, the execution of a task description allows a stakeholder to achieve a goal and the system to reach a steady state.

On the one hand, stakeholders' goals are the execution of the business processes, which allow strategic and operational goals of an organization to be achieved [3]. On the other hand, we consider that the steady states in a PAIS are the same as the ones in the BPDs that will be part of the system and will be determined by those sequences of flow objects that denote tasks that are executed consecutively. For example, the whole business process shown in Fig. 2 corresponds to a task description whose execution will allow the business process and the system to reach a steady state. If the complete sequence of elements is not executed, a steady state will not be reached.

Task descriptions are specified in a textual template that includes the following information: its name; the tasks and the business process that are supported; the role responsible for its execution; the triggers, preconditions and postconditions of the task description; the input and output data objects and their states; an abstract description of the interaction between a user and the PAIS through user intention and system responsibility; the alternatives and extensions to this interaction (which were not taken into account in the business process-based approach); and the business rules that affect the task description. An example of a textual template is not shown, but examples of the previous version of the template (which is identical to the new version except for the alternatives and extensions) are shown in [4] and [5].

Task descriptions are ordered according to their occurrence. Fig. 3 shows the order of the task descriptions that are related to car lifecycle for the rent-a-car company.

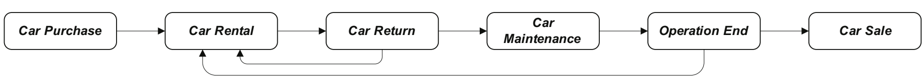


Fig. 3. Example of sequence of task descriptions

3.2 Data Modelling

The second stage of the new approach is data modelling. First, the information flows of each task description are specified on the basis of the domain entities that will be part of the system and the possible interactions that are specified in a task description. Second, a DCD is derived by following a set of guidelines. The DCD contains classes and their attributes, methods and associations.

3.2.1 Specification of Information Flows

System analysts specify the pieces of information that a PAIS and its actors exchange for the execution of its task descriptions by means of information flows. They are specified on the basis of the BNF grammar shown in Fig. 4, which is an adaptation of the way of specifying information flows in the info cases approach. The complete grammar is not shown due to page limitations.

```
<Information flow> ::= <Input flow> | <Output flow> | <Input flow> <Output flow>
<Input flow> ::= → <Data expression>
<Output flow> ::= ← <Data expression>
<Data expression> ::= <Domain entity> | <Domain entity> / <Attribute> /  |
          <Data expression> + <Data  expression> | <Data expression> '|' <Data expression> |
          <Lower limit> { <Data expression> } <Upper limit> | ( <Data expression> ) |
          [ <Data expression> ]
<Attribute> ::= <Attribute name> | <Attribute> + <Attribute> | <Attribute> '|' <Attribute> |
          ( <Attribute> ) | [ <Attribute> ]
```

Fig. 4. Excerpt of the BNF grammar for the specification of information flows

The semantics of the symbols that can appear in an information flow is as follows: the symbols '→' and '←' depict input and output pieces of information to and from a PAIS, respectively; '/ /' depicts membership; '+' depicts composition; '|' depicts alternative; '{ }' depicts repetition; '( )' depicts grouping; and '[ ]' depicts option.

It is essential that the information flows of the task descriptions of a PAIS allow system analysts to completely and correctly identify the elements of a DCD. These elements will be those that are needed for the execution of the task descriptions, and thus for the support of the business processes. In addition, precisely specifying the information flows from a BNF grammar allows the automation of the derivation of a DCD to be possible, although not completely.

Unlike the info cases approach, the new approach carries out the specification of information flows from the domain entities that are used as input and output of task descriptions. Composed, alternative, repeated, grouped and optional elements are based on the normal, alternative and extension interactions of the task description for which an information flow is specified (for brevity, the way to carry out this specification is not explained in detail). Membership elements, which refer to attributes of the domain entities, must be obtained from stakeholders and organizational documentation.

Examples of information flows for the case study are shown in Fig. 6, in which domain entities of the domain data model (Fig. 5) are depicted in bold and in italics.

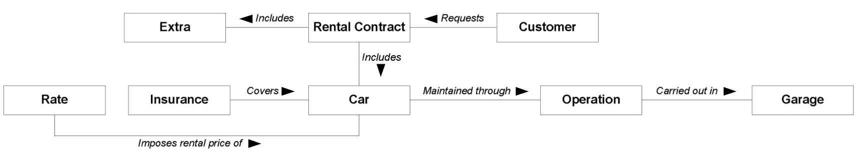


Fig. 5. Example of a domain data model

3.2.2 Modelling of the Domain Class Diagram

At last, the DCD of a PAIS is modelled from the information flows of its task descriptions by following a set of guidelines. The guidelines have been adapted from the rules that are proposed in the info cases approach for the derivation of a domain model. There are 9 guidelines, and they allow system analysts to model the classes of a DCD (guideline 1) and their attributes (guideline 2), methods (guidelines 3, 4, 5 and 6) and associations (guidelines 7, 8 and 9).

Task Description: **CAR PURCHASE**
→ ***Insurance*** + ***Rate*** + ***Car*** / number of plate + model + engine + colour + seats + purchase date + office /
Task Description: **CAR RENTAL**
→ ***Car*** + (***Customer*** | ***Customer*** / number + name + surname + ID number + address + city + telephone number + credit card type + credit card number + credit card expiration date /) + ***Rental Contract*** / contract number + current date + current time + office + return date + return office + [deposit] / + [$_0${ ***Extra*** }$_n$] ← ***Rental Contract*** / contract number + current date + current time + office + return date + return office + rental cost + extras cost + VAT + deposit + total cost / + ***Car*** / model + plate number / + ***Customer*** / name + surname + ID number / + [$_0${ ***Extra*** / name / }$_n$]
Task Description: **CAR RETURN**
→ ***Rental Contract*** / return date / ← ***Rental Contract*** / amount to pay /
Task Description: **CAR MAINTENANCE**
→ ***Car*** + ***Garage*** + ***Operation*** / number + current date + description /
Task Description: **OPERATION END**
→ ***Operation*** / end date + price /
Task Description: **CAR SALE**
→ ***Car*** / sale date /

Fig. 6. Examples of information flows

The guidelines are presented in detail below. We consider that a detailed description is essential given that the DCD of a PAIS is the main outcome of the new approach.

Guideline 1 (classes): A class is modelled in a DCD for each domain entity of an information flow.

For the case study, the classes are "Insurance", "Rate", "Car", "Customer", "Rental Contract", "Extra", "Garage" and "Operation".

Guideline 2 (attributes): An attribute is modelled in a class of a DCD for each attribute that belongs (membership) to the domain entity from which the class was modelled and that is in an input flow; a data type must be specified for each attribute.

For the case study, the attributes of the class "Operation" are "number", "current date", "description", "end date", and "price".

Guideline 3 (creation method): A creation method is modelled for each class of a DCD; its parameters are the attributes of the domain entity from which the class was modelled in the first task description where the domain entity appears; a data type must be specified for each parameter.

For the case study, the creation method of the class "Operation" is "create operation (number, current date, description)".

Guideline 4 (deletion method): A deletion method is modelled in a class of a DCD if: 1) there exists a task description in which the domain entity from which the class was modelled is part of the input flow; 2) the domain entity does not have attributes; 3) the domain entity does not appear in the information flow of any later task description; and 4) the system analyst can confirm that the domain entity is no longer needed.

For the case study, this guideline is not applied.

Guideline 5 (modification method): A modification method is modelled in a class of a DCD for each task description in which the domain entity from which the class was modelled has attributes in an input flow, and the creation method of the class was not modelled from the task description; its parameters are the attributes of the domain entity from which the class was modelled in the task description where the domain entity appears; a data type must be specified for each parameter.

For the case study, a modification method of the class “Operation” is “end operation (end date, price)”.

Guideline 6 (calculation method): A calculation method is modelled in a class of a DCD for each attribute that: 1) belongs to the domain entity from which the class was modelled; 2) is in an output flow; and 3) does not correspond to an attribute of the class; a return data type must be specified for each calculation method.

For the case study, a calculation method of the class “Rental Contract” is “calculate rental cost ()”.

Guideline 7 (associations): An association between two classes of a DCD is modelled if: 1) the domain entities from which the classes were modelled are part of the same input or output flow; and, 2) there exists an association between the entities in the domain data model.

For the case study, the classes “Car” and “Rental Contract” are associated.

Guideline 8 (minimum multiplicity): The minimum multiplicity of a class in an association is 0 if the association is not modelled from the task description from which the creation method of the class was modelled; otherwise, the minimum multiplicity is the minimum number of occurrences of the domain entity from which the class was modelled in the information flow from which the association was modelled (1, 0 if optional, or lower limit of repetitions).

For the case study, the minimum multiplicities of the association “Car – Rental Contract” are 0 for the class “Car” and 1 for the class “Rental Contract”.

Guideline 9 (maximum multiplicity): The maximum multiplicity of a class in an association is the maximum number of occurrences of the domain entity from which the class was modelled in the information flow from which the association was modelled (1 or upper limit of repetitions); the maximum multiplicity could be increased on the basis of business rules.

For the case study, the maximum multiplicities of the association “Car – Rental Contract” are indeterminate (‘*’) for the class “Car” (based on business rules) and 1 for the class “Rental Contract”.

Fig. 7 shows the DCD that has been derived from the information flows shown in Fig. 6 and the domain data model shown in Fig. 5. Note that the information flows are just a part of the whole case study, and thus the DCD is incomplete. In addition, parameters and data types have not been modelled to keep Fig. 7 as small as possible.

Once the guidelines have been presented, two important aspects of a DCD must be pointed out. First, it is evident that a DCD and a domain data model are very similar in the new approach. The classes and associations of the DCD of a PAIS are a subset of the entities and relationships of the domain data model of the organization for which the system will be developed. Nonetheless, we do not consider this fact to be a weakness or problem of the new approach since this is a reflection of common practice in IS development.

The pieces of information (data) that are stored in an IS correspond to a part of the application domain that will be controlled by the system. In the new approach, a domain data model is a part of the application domain that is later analysed and refined

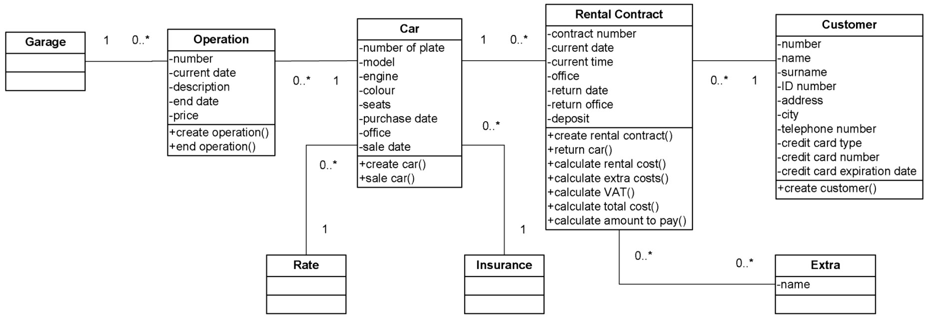


Fig. 7. Example of a domain class diagram

to model a DCD that depicts the part of the domain data that will be controlled by a PAIS. The new approach aims to provide methodological guidance for system analysts in order to model a complete and correct DCD from a domain data model and task descriptions. Furthermore, the purposes of a domain data model and of a DCD are different in the new approach. The purpose of a domain data model is to understand the application domain, whereas the purpose of a DCD is to model the data that will be controlled by a PAIS.

Second, it is known that any class diagram must be complemented with the textual specification of the data constraints that could not be modelled graphically. In the new approach, these constraints are usually specified in the business rules that affect task descriptions, but they might also be specified in a DCD depending on the preferences of system analysts.

4 Conclusions and Future Work

Organizational and business process modelling as a means for understanding the application domain are essential for PAIS development. BPDs play a major role in PAIS modelling, but system analysts must not limit their focus on them alone. They must also take other aspects such as data models into account.

This paper has presented a RE approach that provides methodological guidance to help system analysts model the data of a PAIS. The new approach is based on two other RE approaches whose integration can be regarded as the main contribution of this paper. BPMN has been extended with consecutive flow, task descriptions have been improved though the extension of their textual template and the adoption of a homogeneous abstraction level, and new guidelines for the specification of information flows and for DCD modelling have been presented.

The integration takes advantage of the strong points of the two approaches and also extends them. The business process-based approach now addresses data modelling, and the info cases approach now addresses organizational modelling for the elicitation of functional requirements.

As future work, the development of tool support is planned to facilitate the use of the new approach. A technique for the analysis of non-functional requirements and guidelines for the derivation of the presentation model (user interface) of OO-Method are also necessary. Lastly, the new approach must be applied in more projects in order to further evaluate it and so that improvements might be made. It is important that the new approach is used in large projects and in projects in which a legacy system exists.

Acknowledgements. The authors would like to thank the reviewers for their useful comments and suggestions for the paper.

References

1. Alexander, I., Bider, I., Regev, G.: REBPS 2003: Motivation, Objectives and Overview. Message from the Workshop Organizers. In: Eder, J., Missikoff, M. (eds.) CAiSE 2003. LNCS, vol. 2681. Springer, Heidelberg (2003)
2. Bubenko, J., Persson, A., Stirna, J.: EKD User Guide (2001), `http://www.dsv.su.se/~js`
3. de la Vara, J.L., Sánchez, J.: Business process-driven requirements engineering: a goal-based approach. In: BPMDS 2007, Trondheim, Norway (2007)
4. de la Vara, J.L., Sánchez, J.: Impoving Requirements Analysis through Business Process Modelling: A Participative Approach. In: Abramowicz, W., Fensel, D. (eds.) BIS 2008. LNBIP, vol. 7, pp. 165–176. Springer, Heidelberg (2008)
5. de la Vara, J.L., Sánchez, J., Pastor, Ò.: Business process modelling and purpose analysis for requirements analysis of information systems. In: Bellahsène, Z., Léonard, M. (eds.) CAiSE 2008. LNCS, vol. 5074, pp. 213–227. Springer, Heidelberg (2008)
6. Díaz, I., Sánchez, J., Matteo, A.: Conceptual Modelling Based on Transformation Linguistic Patterns. In: Delcambre, L.M.L., Kop, C., Mayr, H.C., Mylopoulos, J., Pastor, Ó. (eds.) ER 2005. LNCS, vol. 3716, pp. 192–208. Springer, Heidelberg (2005)
7. Dumas, M., van der Aalst, W., ter Hofstede, A. (eds.): Process-Aware Information Systems. Wiley, Chichester (2005)
8. Fortuna, M.H., Werner, C.M.L., Borges, M.R.S.: Info Cases: Integrating Use Cases and Domain Models. In: RE 2008, Barcelona, Spain (2008)
9. Glinz, M.: A Lightweight Approach to Consistency of Scenarios and Class Models. In: ICRE 2000, Schaumburg, USA (2000)
10. Insfran, E., Pastor, Ó., Wieringa, R.: Requirements Engineering-Based Conceptual Modelling. Requirements Engineering 7(2), 61–72 (2002)
11. Kösters, G., Six, H.W., Winter, M.: Coupling Use Case and Class Models as a Means for Validation and Verification of Requirements Specifications. Requirements Engineering 6(1), 3–17 (2001)
12. Nigam, A., Caswell, N.S.: Business artifacts: An approach to operational specification. IBM Systems Journal 42(3), 428–445 (2003)
13. OMG: Business Process Modelling Notation (BPMN) Specification v1.1, (2008), `http://www.bpmn.org`
14. Pastor, O., Molina, J.C.: Model-Driven Architecture in Practice. Springer, Heidelberg (2007)
15. Reijers, H.A., Liman, S., van der Aalst, W.M.P.: Product-Based Workflow Design. Journal of Manegement Information Systems 20(1), 229–262 (2003)
16. Rodriguez, A., Fernández-Medina, E., Piattini, M.: CIM to PIM Transformation: A Reality. In: CONFENIS 2007, Pekin, China (2007)
17. Scheer, A.W.: ARIS - Business Process Modeling, 3rd edn. Springer, Heidelberg (2000)
18. Shun, S.X., Zhao, J.L., Numaker, J.F.: Formulating the Data-Flow Perspective for Business Process Management. Information Systems Research 17(4), 374–391 (2006)
19. Siau, K., Lee, L.: Are use case and class diagram complementary in requirements analysis? Requirements Engineering 9(4), 229–237 (2004)
20. Svetinovic, D., Berry, D.M., Godfrey, M.: Concept Identification in Object-Oriented Domain Analysis: Why some Students Just Don't Get It. In: RE 2005, Paris, France (2005)
21. Wang, J., Kumar, A.: A Framework for Document-Driven Workflow Systems. In: van der Aalst, W.M.P., Benatallah, B., Casati, F., Curbera, F. (eds.) BPM 2005. LNCS, vol. 3649, pp. 285–301. Springer, Heidelberg (2005)

Defining Adaptation Constraints for Business Process Variants

Ruopeng Lu[1], Shazia Sadiq[2], Guido Governatori[3], and Xiaoping Yang[2]

[1] SAP Research CEC Brisbane, Australia
ruopeng.lu@sap.com
[2] School of Information Technology and Electronic Engineering
The University of Queensland, Brisbane, Australia
{shazia,xiaoping}@itee.uq.edu.au
[3] NICTA, Australia
guido.governatori@nicta.com.au

Abstract. In current dynamic business environment, it has been argued that certain characteristics of ad-hocism in business processes are desirable. Such business processes typically have a very large number of instances, where design decisions for each process instance may be made at runtime. In these cases, predictability and repetitiveness cannot be counted upon, as the complete process knowledge used to define the process model only becomes available at the time after a specific process instance has been instantiated. The basic premise is that for a class of business processes it is possible to specify a small number of essential constraints at design time, but allow for a large number of execution possibilities at runtime. The objective of this paper is to conceptualise a set of constraints for process adaptation at instance level. Based on a comprehensive modelling framework, business requirements can be transformed to a set of minimal constraints, and the support for specification of process constraints and techniques to ensure constraint quality are developed.

Keywords: Business Process Management, Process Variant Management, Constraint-Based BPM, Business Process Constraint Network.

1 Background and Motivation

In order to provide a balance between the opposing forces of control and flexibility, we have argued for [12], a modelling framework that allows part of the model that requires less or no flexibility for execution to be predefined, and part to contain loosely coupled process activities that warrant a high level of customization. When an instance of such a process is created, the process model is *concretised* by the domain expert at runtime. The loosely-coupled activities are given an execution plan according to instance-specific conditions, possibly some invariant process constraints, and their expertise.

The foremost factor in designing business processes is achieving improvements in the business outcomes [4]. However, decisions at the strategic level need to

W. Abramowicz (Ed.): BIS 2009, LNBIP 21, pp. 145–156, 2009.

be evaluated in light of constraints that arise from several sources. It has been identified that at least four sources of constraints have impact on a business process design:

- **Strategic constraints** define the tactical elements of the process e.g. approval of director required for invoices beyond a certain value.
- **Operational constraints** are determined through physical limitations of business operations, e.g. minimum time for warehouse offloading.
- **Regulatory constraints** are prescribed by external bodies and warrant compliance e.g. financial and accounting practices (Sarbanes-Oxley Act), or batch identification for FDA in pharmaceutical industry.
- **Contractual constraints** define the contractual obligations of an organization towards its business partners, e.g. maximum response time for a service.

In order to harness the full power of BPM techniques, each of these constraints should eventually be translated into constructs of a (executable) business process model, and subsequently lead to process enforcement at the business activity level. This paper will introduce a *design time* modelling approach for *process constraints*. This approach transfers part of the process modelling effort to domain experts who make execution decisions at runtime. Instance adaptation is supported by techniques for specifying instance-specific process models and constraint checking in different variants of the business process [12]. We will demonstrate how the specification of so-called selection constraints can lead to increased flexibility in process execution, while maintaining a desired level of control.

The rest of the paper is organized as follows. We use a running example to motivate the overall approach in section 2. Section 3 provides background concepts for the underlying framework for supporting instance adaptation. In section 4, we introduce the core concept of process constraints. The approach is evaluated against an application scenario in section 5, where the constraint editor prototype is presented. Related work is presented in section 6, followed by the conclusion and future work in section 7.

2 Motivating Example

Consider a customer request processing workflow in a CRM (Customer Relationship Management) system. The process is triggered when a customer submits a purchase request. Upon receiving which, a senior sales representative creates a customer requirement document in the system, then he specifies a list of activities to manage the customer request for different workflow roles/participants. The purchase request is finally confirmed and further procurement activities are carried out. The procedures for creating and confirming customer requirement document are predictable and repetitive, while the customer request management activities will typically be case-specific and can be uniquely configured for each customer request which require ad-hoc adjustment. Fig. 1 shows such a

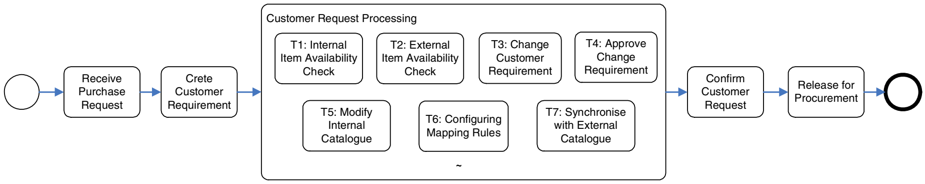


Fig. 1. A network diagnostics scenario modelled in BPMN notation

workflow in BPMN notation, where the dynamic part of the workflow is modelled as an Ad-Hoc Sub-Process [9].

At the same time, some dependencies and restrictions among the available tests can be abstracted beforehand as follows:

- Internal Item Availability Check $T1$ must be performed for all customer requirement cases;
- Based on operational guideline, maximal 6 activities can be selected for prompt customer response;
- The customer requirement can be adjusted according to case-specific conditions ($T3$), but needs to be approved by a senior sales representative ($T4$);
- The mapping rules can be configured to map content from external catalogue to internal catalogue ($T6$), after which the internal catalogue should be synchronised with the external catalogue to reflect the rule change ($T7$). However, synchronisation $T7$ can be triggered without change in mapping rules;
- In order to avoid inconsistency, manual modification to internal catalogue ($T5$) and automatic synchronisation with external catalogue ($T7$) should not be selected at the same workflow instance.

It is obvious that even with a small number of activities, there is a large number of possible scenarios that satisfy the selection restrictions. For example, the sales representative can decide to execute one, two or up to six activities to fulfil the request processing goals, where the combination of selected activities can be varied. Consider that for each such scenario, the sales representative can further define a variety of execution plans including any execution order and patterns, e.g., execute all selected tasks in sequence, in parallel, or some in sequence and others in parallel (Fig. 2). In graphical modelling notation (e.g., BPMN), it is tedious if not possible to capture and manage all such scenarios and to articulate the conditions for selecting the alternative branches.

3 Fundamentals

Constraint Satisfaction are supported by formal methods and its suitability and applicability to solve applied problems such as job-shop scheduling have been extensively demonstrated [2]. The declarative nature of constraint specification,

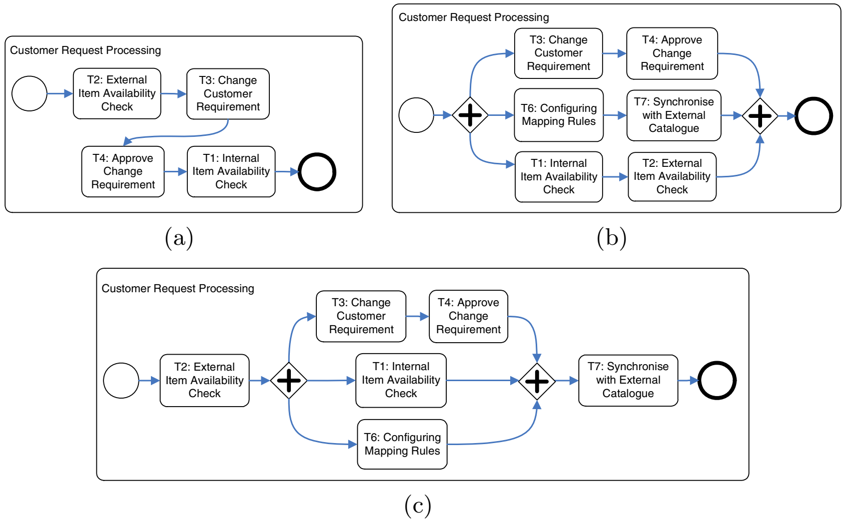


Fig. 2. Execution possibilities of the BPMN ad hoc sub-process shown in Fig. 1

and the automated reasoning backed by formal theory provides a potential hand-in-gloves solution for the flexible workflow problem defined in this paper.

Nevertheless, applying Constraint Satisfaction in flexible workflows is not straightforward. The overall challenge is to find the right balance between the expressiveness of the constraint language that is rudimentary for expressing constraints in workflows, and the computational efficiency of the resulting constraint networks. The fundamental requirements are to design a constraint language that is expressive enough, while computationally efficient algorithms can be designed and applied for reasoning, and is usable for human constraint designer.

Business process semantics is complex in a sense that it involves different level of abstractions. Control flow, data dependency and resource allocation are the primary aspects that needs to be modelled, among which control flow is the fundamental aspect in workflow models [5]. As such, process constraints should express foremost and at least the restrictions on the structural dependency between workflow activities in a workflow model.

Through intensive investigations, a variant of Constraint Satisfaction have been identified as the theoretical foundation for process constraints, namely, Boolean Constraint Network. Boolean constraints are used to specify the inter-task dependencies, e.g., the effect on other tasks when selecting task A to execute, and should task B be selected if A is depending on B (for logical or data dependency). In particular, the consideration of Boolean constraints is confined within the subset of *tractable* constraints. Tractable constraints are the the class of constraints where polynomial time solution algorithms are known [3]. A mapping has been established between process requirements as directed by business process semantics, and the formal expressions in a subset of Boolean constraints. Then the expressiveness of the constraint language can be determined, which justifies the appropriateness of the constraint language for expressing process requirements. Once the constraint language is defined, appropriate algorithms can be applied for constraint validation, and process design verification.

Definition 1 (Constraint). *A constraint* C *is a pair* (L, R)*, where* $L \subseteq X$ *is an* m*-tuple of variables, called the scope of the constraint;* X *is a finite set of variables.* R *is an* m*-ary relation defined on* L*, which is a subset of the set of all* m*-tuples of elements from some set* D*, called the domain.*

The set of all m-tuples $(d_1, d_2, \ldots, d_k)$ of elements from some set D with $d_1 \in D_1, d_2 \in D_2, \ldots, d_k \in D_k$ is the *Cartesian* product $D_1 \times D_2 \times \ldots \times D_k$. The length of the tuple m is called the arity of the constraint. When there is no ambiguity on the scope of the constraints, a relation and a constraint is referred to interchangeably.

For the subsequent discussion on process constraints, we also define this specific term: A process template PT is defined by a set of process tasks T and a set of process constraints C. T represent the available pool of activities to be adapted at runtime. C is a set of selection constraints that defines relations between the properties of tasks in T.

4 Selection Constraints

The adaptation of a process instance is governed by selection and subsequent scheduling constraints. Selecting constraints regulates how and what tasks can be chosen to perform, while scheduling constraints address how these selected tasks are executed, e.g., order of execution, in sequence or parallel [7]. This paper focuses on the first perspective.

4.1 Conceptualisation

The following classes of selection constraints have been identified:

- **Mandatory.** Constraint *man* defines a set of tasks that *must be executed* in every process variant, in order to guarantee that intended process goals will be met.
- **Prohibitive.** Constraint *pro* defines a set of tasks that *should not be executed* in any process variant.
- **Cardinality.** Constraint specifies the minimal *minselect* and maximal *maxselect* cardinality for selection among the set of available tasks.
- **Inclusion.** Constraint *inc* expresses the *dependency* between two tasks T_x and T_y, such that the presence of T_x imposes restriction that T_y must also be included. Prerequisite constraint *pre* is the inverse of an inclusion constraint.
- **Exclusion.** Constraint *exc* prohibits T_y from being included in the process variant when the T_x is selected.
- **Substitution.** Constraint *sub* defines that if T_x is not selected, then T_y must be selected to *compensate the absence* of the former.

Corequisite. Constraint *cor* expresses a stronger restriction in that either both T_x and T_y are selected, or none of them can be selected, i.e., it is not possible to select one task without the other.

Exclusive-Choice. Constraint *xco* is also a more restrictive constraint on the selection of alternative tasks, which requires at most one task to be selected from a pair of tasks (T_x, T_y).

4.2 Formalisation

Let $T = \{T_1, T_2, \ldots, T_n\}$ denote the set of all tasks in a process template PT. Each task T_i is considered as a propositional variable ranging over domain $D_i = \{0, 1\}$. Let $T_i = 1$ stand for the *presence* of task T_i in a process variant V and $T_i = 0$ stand for *absence*.

Mandatory, prohibitive and cardinality constraints can be defined by restricting the domains of respective tasks. A mandatory task $T_i \in T$ is denoted by $man(T_i)$, where *manselect* is a property of T_i restricting its domain $D_i = \{1\}$. The set of all mandatory tasks in a process template PT is given by:

$$R^{man} = \{T_i \mid man(T_i)\}$$

A task T_i in a process template PT is prohibitive if it is forbidden to be selected in any process variants V of PT. A single prohibitive task T_x can be denoted by $pro(T_x)$, where *pro* is a property of T_x restricting $D_x = \{0\}$. The set of all prohibited tasks in a process template PT is given by:

$$R^{pro} = \{T_i | \; pro(T_i)\}$$

A *minselect* constraint is denoted by $R^{min(m)} \subseteq T$, such that $|R^{min(m)}| \geq m$, and $\forall T_i \in R^{min(m)}, D_i = \{1\}$. The *minselect* constraint restricts that every process variant V should contain all tasks in R^{man}, and zero, one or more tasks from $(T - R^{man})$. A *maxselect* constraint is denoted by $R^{max(m)} \subseteq T$, such that $|R^{max(m)}| \leq m$, and $\forall T_i \in R^{max(m)}, D_i = \{1\}$.

The mandatory, prohibitive and cardinality constraints are defined by restricting the domain of a single task. On the other hand, inclusion, exclusion, substitution, prerequisite, corequisite and exclusive-choice constraints are binary relations that are defined by restricting the domains of the pair of tasks. According to the nature of the constraints, we call them **containment constraints**. For example, An inclusion constraint R^{inc} is a binary relation on a pair of variables (tasks) $T_x, T_y \in T$, if and only if (*iff*):

$$R^{inc} = ((T_x, T_y), \{(0,0), (0,1), (1,1)\})$$

An inclusion constraint R^{inc} defined on tasks T_x, T_y reads T_x includes T_y. By definition, it restricts the domain of values that can be assigned to the pair (T_x, T_y). In this case, either $(0,0)$, $(0,1)$, or $(1,1)$ can be assigned. Applying this definition to task selection, it expresses that when T_x is selected, T_y must also be selected (T_x is the dependent of T_y). The following selection scenarios are permitted:

- neither T_x nor T_y is selected, i.e., (0, 0);
- T_y is selected without T_x, i.e., (0, 1);
- both T_x and T_y are selected, i.e., (1, 1).

The scenario $(1, 0)$ is prohibited where T_x is selected without T_y, thus enforcing the inclusion relationship between selection of T_x and T_y.

Similarly, an exclusion constraint R^{exc} is a binary relation on a pair of variables $T_x, T_y \in T$, *iff*:

$$R^{exc} = ((T_x, T_y), \{(0, 0), (0, 1), (1, 0)\})$$

An exclusion constraint prohibits the selection scenario $(1, 1)$ where both T_x and T_y are selected. Table 1 presents a summary for the definition of the containment constraints.

Table 1. Definitions of Containment Constraints

Constraint	Definition
R^{inc}	$((T_x, T_y), \{(0, 0), (0, 1), (1, 1)\})$
R^{exc}	$((T_x, T_y), \{(0, 0), (0, 1), (1, 0)\})$
R^{sub}	$((T_x, T_y), \{(0, 1), (1, 0), (1, 1)\})$
R^{pre}	$((T_x, T_y), \{(0, 0), (1, 0), (1, 1)\})$
R^{cor}	$((T_x, T_y), \{(0, 0), (1, 1)\})$
R^{xco}	$((T_x, T_y), \{(0, 1), (1, 0)\})$

For example, the restrictions for customer request processing discussed in section 2 can be expressed as follows:

- *Mandatory* task $T1$, i.e., $man(T1)$
- *Maximal* selection of 6 tasks, i.e., $R^{max(6)}$
- $T3$ and $T4$ are *co-requisite*, i.e., $T3$ *cor* $T4$
- $T6$ *includes* $T7$, i.e., $T6$ *inc* $T7$
- $T5$ and $T7$ are *exclusion*, i.e., $T5$ *exc* $T7$

Considering constraint $T3$ *cor* $T4$, where it regulates that either both $T3$ and $T4$ are selected in the same workflow instance, or none of them should be. Similarly, the substitution constraint $T5$ *sub* $T7$ prohibits the case when both $T5$ and $T7$ are selected at the same instance.

4.3 Expressiveness of Selection Constraints

A constraint language L is defined by imposing restrictions on the possible constraint relations allowed to use. The expressiveness of the selection constraint language can be investigated through the inverse, intersection and composition of the abovementioned selection constraints. These constraint properties will also be utilised in the validation algorithm later in the paper.

Inverse. Given a containment constraint R, the ***inverse*** R^{-1} of R also follows basic set theoretic definition, i.e.,

$$R^{-1} = \{(a,b)|(b,a) \in R\} \tag{1}$$

The inverse of a prerequisite constraint is equivalent to an inclusion constraint. The rest of the four containment constraints are reflexive, therefore their inverse relations equal to themselves. The inverse of a mandatory constraint is the prohibitive constraint and vice versa.

Intersection. For (binary) containment constraints, it is allowed for more than one selection constraint to be specified on the same pair of tasks (variables). The ***intersection*** between two containment constraints R_1 and R_2 defined on the same pair of variables (T_x, T_y) is denoted by $R_1 \cap R_2$. Intersections between selection constraints follow the classic set theoretic definition, i.e.,

$$R_1 \cap R_2 = \{(a,b)|(a,b) \in R_1,\ (a,b) \in R_2\} \tag{2}$$

For example, suppose an inclusion and a substitution constraint are defined on tasks T_x and T_y, i.e., T_x *inc* T_y, and T_x *sub* T_y. The set of possible values for (T_x, T_y) that satisfies both *inc* and *sub* is given by:

$$inc \cap sub = \{(0,0),(0,1),(1,1)\} \cap \{(0,1),(1,0),(1,1)\} = \{(0,1),(1,1)\}$$

The resulting relation requires T_y to be a mandatory task since $T_y = 1$ in both cases. Furthermore, the intersection between containment constraints and the mandatory and prohibitive constraints can also be defined.

Composition. The ***composition*** of two containment constraints $R(T_x, T_y)$ and $R(T_y, T_z)$, denoted by:

$$R(T_x, T_y) \otimes R(T_y, T_z)$$

results in the binary relation $R(T_x, T_z)$, where

$$\begin{aligned} R(T_x, T_z) &= R(T_x, T_y) \otimes R(T_y, T_z) \\ &= \{(a,b)|a \in D_x, b \in D_z, \exists c \in D_y, (a,c) \in R(T_x, T_y), (c,b) \in R(T_y, T_z)\} \end{aligned} \tag{3}$$

In order to make mandatory and prohibitive relations union-comparable to containment relations, the relation $R(T_x, T_y)$ between an unconstrained task, say T_x, and a mandatory task T_y can be denoted as $\{(0,1),(1,1)\}$, which allows any value of T_x as long as $T_y = 1$. Similarly, if T_y is prohibitive, and T_x is unconstrained, $R(T_x, T_y) = \{(0,0),(1,0)\}$. For example, suppose defining T_x *pre* T_y on an arbitrary pair of tasks (T_x, T_y), while T_y is mandatory. It implies that both T_x and T_y to be mandatory, i.e.,

$$R(T_x, T_y) = \{(0,0),(1,0),(1,1)\} \cap \{(0,1),(1,1)\} = \{(1,1)\}$$

While if T_x is prohibitive, and T_x *pre* T_y for tasks T_x and T_y, it can be inferred that T_y is also prohibitive.

4.4 Validation of Selection Constraints

Given the definition of selection constraints in a process template PT, it is necessary to make sure at design time that:

- all implicit constraints are made explicit; and
- no conflict exists in the constraint specification.

In order to formally address the problem, a Selection Constraint Network (SCN) is defined to provide formal semantics to validate the selection constraints. SCN is a binary Boolean constraint network [6], which is a special type of constraint network with variables having bi-valued domains 1, 0 or true, false. Validation of selection constraints of a process template PT is realized by checking for the consistency of the corresponding SCN.

Definition 2 (SCN). *Given a process template PT, SCN is a binary constraint network defined by (X^s, D^s, C^s). $X^s = \{T_1, \ldots, T_n\}$ is the set of all tasks in PT represented as propositional variables. $D^s = \{D_1, \ldots, D_n\}$ is the set of domains of the corresponding propositional variables, where $D_i \in D^s, D_i \subseteq \{0, 1\}$. C^s is the set of all selection constraints in PT.*

For example, a $SCN = (X^s, D^s, C^s)$ can be formulate for the the selection constraints modelled for the customer request processing workflow in section 4, where:

- $X^s = \{T1, \ldots, T7\}$
- $D^s = \{D1, \ldots, D7\}$, where

$$Di = \begin{cases} \{1\} & i \in \{1\} \\ \{0,1\} & i \in \{2,3,4,5,6,7\} \end{cases}$$

- $C^s = \{R(T3, T4), R(T5, T7), R(T6, T7)\}$, where
  - $R(T3, T4) = T3\ cor\ T4$,
  - $R(T5, T7) = T5\ inc\ T7$, and
  - $R(T6, T7) = T6\ exc\ T7$

A *solution* to a SCN is the assignment of values to each variable such that no constraint is violated, i.e., $\forall T_i \in X^s$, there is a consistent assignment of a value from D_i such that all relevant selection constraints in C^s are satisfied. A solution corresponds to a task selection scenario for the process template PT that satisfies all relevant selection constraints in PT. A SCN is said to be *consistent* if there exists at least one solution. If conflict exists between the constraints, the constraint network is inconsistent and hence no solution exists.

Given an SCN as a binary Boolean constraint network, the consistency of SCN can be checked by applying a generic consistency checking algorithm to SCN (cf. Fig. 3 adapted from [3]).

The algorithm checks whether each pair of constraints in SCN is consistent. If conflicts exist, the operation $R(T_i, T_j) \cap (R(T_i, T_k) \otimes R(T_k, T_j))$ results in an

Procedure *SCN Consistency*
Input: A constraint network $SCN\ (X^s, D^s, C^s)$
Output: A consistent constraint network SCN' equivalent to SCN
Method:

```
do
  Q ← SCN
  for each k from 1 to n do
    for each i, j from 1 to n, R(T_i, T_j) ∈ Q do
      R(T_i, T_j) ← R(T_i, T_j) ∩ (R(T_i, T_k) ⊗ R(T_k, T_j))
      if R(T_i, T_j) = ∅ then break
until Q = SCN
```

Fig. 3. Consistency checking procedure for SCN

empty constraint $R(T_i, T_j)$ for some T_i and T_j, and the checking algorithm terminates. Conflicts can exist between containment constraints, or between mandatory and containment constraints. However, if no conflict is detected, further composition operation will not change the constraints in SCN, and the algorithm terminates. Furthermore, implicit constraints will be inferred and added to SCN, as the result of applying the composition operation to each pair of tasks.

Considering the following example to help explain how the algorithm is applied to reason about the selection constraints. Applying the algorithm to the given SCN example, a more restrictive relation between $(T5, T6)$ is obtained by applying composition $\otimes$ to $R(T5, T7)$ and $R(T6, T7)$. We take the inverse of $R(T6, T7)$, where $R^{-1}(T6, T7) = R(T7, T6) = T7\ pre\ T6$. Applying composition operation $R(T5, T7) \otimes R(T7, T6)$, the new relation $R(T5, T6) = T5\ exc\ T6$ is obtained (cf. equations (1) and (3) in section 4.3). Since no conflict is detected, the example SCN is consistent[1]. Hence, the specification of selection constraints for the customer request processing workflow is correct.

5 Prototypical Implementation

We have implemented a prototype to evaluate the theoretical constraint modelling approach. An extensible constraint designer has been built for end users to model selection constraints (and other constraint types such as scheduling constraints [7]). Fig. 4 is a snapshot of the prototype showing the modelled constraints from the previous examples. A built-in function is provided for design time constraint validation. If validation is successful, the editor exports the constraint specification into a XML file, which can be deployed into a dynamic workflow engine called Chameleon. Chameleon is a light-weight workflow platform built upon Windows Workflow Foundation that supports flexible workflow execution such as Pocket of Flexibility [12]. The deployed constraints file is used

[1] Note that the cardinality constraint can be simply checked by counting the number of selected tasks in X^s.

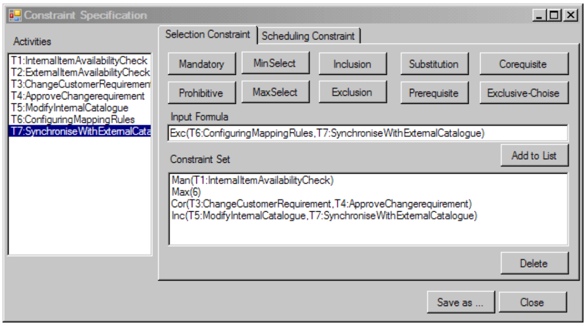


Fig. 4. Constraint Editor Prototype

to automatically verify instance adaptation for different process instances on the Chameleon platform during instance adaptation.

6 Related Work

The proposed constraint-based approach falls in the category of *late binding* paradigm, which is an approach where parts of the process schema are left undefined at design time, but are configured at runtime for each process instance. For this purpose, placeholder activities are provided, which are modelled and executed during run-time. The guidelines or restrictions on the runtime adaptation are modelled as rules [1,8] or constraints [12]. On the other hand, there have been a number of constraint languages proposed in other disciplines, such as Object Constraint Language (OCL) [10] for object-oriented analysis and design method. OCL is based on first-order predicate logic but it uses syntax similar to programming languages. As a result, it is used to express additional constraints on UML models that cannot specified with the graphical means provided by UML. OCL has been applied to model resource allocation in workflows. However OCL and UML do not support concepts usually adopted in the characterisation of control flows requirement in workflows. At the same time, the DECLARE approach [11] has a similar problem-solving philosophy, which also aims at supporting instance-level process adaptation by defining a set of workflow constraints to regulate flexible changes. The background theory of DECLARE constraints is Linear Temporal Logic (LTL), which operates on additional temporal logical operators including *always*, *eventually*, *until* and *next time*. As a result, the focus of DECLARE constraints has been on the temporal dependencies. In our constraint framework, late binding is seen as a two-step approach. At runtime, domain expert adapts a process instance by first selecting a set of tasks to perform, and secondly structuring the tasks into certain workflow patterns. As a result, two constraints systems have been developed to support such activities. This paper has focused on modelling the dependencies on task selection, as the first and most essential step in late binding paradigm. The structuring (scheduling) constraints have been covered by our previous work [7].

7 Conclusion and Outlook

Process constraints can express minimal restrictions on the selection and ordering of tasks for all instances of the targeted business process, thus providing a degree of flexibility in process execution. This paper has presented how task selection constraints can be specified at *design time*, through selection constraints, and the quality of the constraint specification is checked through the formal machinery of selection constraint network respectively. Different process models can be built/tailored for individual process instances at runtime, leading to instance-specific process models (process variants). Possible future work includes to explore further constraint dimensions such as the resource perspective, and a complete implementation of the extended constraint sets.

References

1. Adams, M., ter Hofstede, A.H.M., Edmond, D., van der Aalst, W.M.P.: Implementing Dynamic Flexibility in Workflows using Worklets. BPM center.org. (2006)
2. Barták, R.: Dynamic constraint models for planning and scheduling problems. In: Apt, K.R., Kakas, A.C., Monfroy, E., Rossi, F. (eds.) Compulog Net WS 1999. LNCS, vol. 1865, pp. 237–255. Springer, Heidelberg (2000)
3. Dechter, R.: Constraint Processing. Morgan Kaufmann Publishers, San Francisco (2003)
4. Indulska, M., Chong, S., Bandara, W., Sadiq, S., Rosemann, M.: Major issues in business process management: A vendor perspective. In: Proc. the Pacific Asia Conference on Information Systems (PACIS 2007) (2007)
5. Jablonski, S.: MOBILE: A Modular Workflow Model and Architecture. In: Proc. Int'l. Working Conference on Dynamic Modelling and Information Systems, Nordwijkerhout 1994 (1994)
6. Jeavons, P.: Constructing constraints. In: Maher, M.J., Puget, J.-F. (eds.) CP 1998. LNCS, vol. 1520, p. 2. Springer, Heidelberg (1998)
7. Lu, R., Sadiq, S., Padmanabhan, V., Governatori, G.: Using a Temporal Constraint Network for Business Process Execution. In: Proc. 17th Australasian Database Conference (ADC 2006), Hobart, Australia (2006)
8. Muller, R., Greiner, U., Rahm, E.: AGENT WORK: a workflow system supporting rule-based workflow adaptation. Data Knowl. Eng. 51, 223–256 (2004)
9. Object Management Group: Business Process Modeling Notation (BPMN) Specification 1.0 Technical Report, Object Management Group (OMG) (2006)
10. Object Management Group: Object Constraint Language Specification Version 1.1. Technical Report, Object Management Group (OMG) (1997)
11. Pesic, M., Schonenberg, M.H., Sidorova, N., van der Aalst, W.M.P.: Constraint-based workflow models: Change made easy. In: Meersman, R., Tari, Z. (eds.) OTM 2007, Part I. LNCS, vol. 4803, pp. 77–94. Springer, Heidelberg (2007)
12. Sadiq, S., Sadiq, W., Orlowska, M.: A Framework for Constraint Specification and Validation in Flexible Workflows. Information Systems 30(5) (2005)

MMC-BPM: A Domain-Specific Language for Business Processes Analysis

Oscar González[1,2,⋆], Rubby Casallas[1], and Dirk Deridder[2,⋆⋆]

[1] Universidad de los Andes, TICSw Group, Cra. 1 N∘ 18A 10, Bogotá, Colombia
{o-gonza1,rcasalla}@uniandes.edu.co
[2] Vrije Universiteit Brussel, SSEL, Pleinlaan 2, 1050 Brussel, Belgium
dirk.deridder@vub.ac.be

Abstract. Business Process Management approaches incorporate an analysis phase as an essential activity to improve business processes. Although business processes are defined at a high-level of abstraction, the actual analysis concerns are specified at the workflow implementation level resulting in a technology-dependent solution, increasing the complexity to evolve them. In this paper we present a language for high-level monitoring, measurement data collection, and control of business processes and an approach to translate these specifications into executable implementations. The approach we present offers process analysts the opportunity to evolve analysis concerns independently of the process implementation.

Keywords: Business Process Analysis, Domain Specific Language, Monitoring, Measurement, Control.

1 Introduction

Nowadays business process management (BPM) technologies are frequently used by companies to model, automate, execute and analyze executable business processes [1]. Business processes facilitate the integration of human and technological resources in an organization, according to a set of activities that fulfill certain policy goals. BPM has evolved from traditional workflow management systems (WFMS) including a Business Process Analysis (BPA) phase in its lifecycle. BPA is crucial to recover detailed feedback about how business processes are being executed to facilitate their effective improvement.

BPM evidences the difficulty of analyzing the execution of processes from a business perspective [2]. Firstly, while the process models are automated from a business perspective in a workflow implementation, the analysis specifications are realized from an implementation perspective. Although there are many tools and techniques to analyze business processes (*e.g.,* business activity monitoring) [3] [4],

⋆ Funded by the Flemish Interuniversity Council (VLIR) and COLCIENCIAS "Instituto Colombiano para el Desarrollo de la Ciencia y la Tecnología".

⋆⋆ Funded by the Interuniversity Attraction Poles Programme - Belgian State Belgian Science Policy.

W. Abramowicz (Ed.): BIS 2009, LNBIP 21, pp. 157–168, 2009.

typically these solutions use their own languages to encode analysis concerns in the implementation of the workflow system (*e.g.,* BPEL or java). Secondly, typically these analysis specifications result in tangled and scattered knowledge in the process code. These low-level mechanisms decrease the maintainability and reusability capabilities and increase the complexity because analysis concerns have to be repeated or adapted every time the process change. On the other hand, current BPA solutions typically only perform analysis in terms of process execution information (*e.g.,* time running, current state) but not in terms of the inner definition of basic activities (data). However, it is necessary to be aware immediately when a critical attribute value is assigned in order to take complementary decisions (*e.g.,* allow data modification) (Section 2).

In this paper we present our approach to realize analysis concerns from a business perspective and to automatically add them to an existing process implementation. Our work focuses on supporting the explicit description of the monitoring of business processes, the related measurement data collection, and the definition of control actions over this data. We realize this in such a way that our approach is independent of a particular process implementation language. This is done by working directly on BPMN, which is a high-level standard notation that can be used by process analysts to generalize the analysis specifications independent of specific implementation details. We define a domain-specific language to specify analysis concerns in terms of domain concepts (*i.e.,* BPMN) and a mechanism to express the data used by the process model to improve analysis capabilities. (Section 3). We provide the execution semantics of our language with a suitable aspect-oriented workflow language (Section 4).

The processes analyzed *a posteriori* and other kind of analysis problems such as verification or validation of processes, automatic improvements to the process model, and predictions are out of the scope of this work (Section 5). We conclude by discussing promising pathways for improving our work (Section 6).

2 Business Process Analysis Scenario

BPM supports processes modeling and derivation of executable definitions using multiple languages. We adopted the Business Process Modeling Notation (BPMN) [6] for the business perspective and the Business Process Execution Language (BPEL) [7] for the technical perspective since they are currently the de-facto standard in these areas. BPMN can be interpreted as a domain specific language (DSL) to describe business processes at a conceptual level. It bridges the communication gap that frequently occurs between the design of a business process and its corresponding implementation. However, BPMN suffers from a lack of support to specify data-flow and the implementation of activities required to automate processes [8]. BPMN models can be translated into an executable model such as BPEL, in which process instances are enacted by workflow engines.

This section introduces a Loan Approval Process as a running example [7]. In addition, we present a concrete scenario describing a number of analysis requirements, and the problems that users face to realize such requirements.

2.1 Case Study: A Loan Approval Process

The Loan Approval process starts with a loan request, where the client uses the bank′s webpage to introduce personal information and the amount requested. After this, the process executes three activities to decide if the request is approved or rejected. The first activity in the series is a task that invokes a Web service, which evaluates the information provided by the client. This activity is based on a set of business rules defined by the bank. For example, if the client income is less than 10% of the requested amount then the loan request is rejected; if the amount is less than $5000 then the loan request is approved. The second activity, which is executed in parallel with the previous one, is a subprocess that validates the loan viability by evaluating the risk associated with the requester. This subprocess seeks advice from a credit entity and validates the authenticity of the data provided by the client. The third activity merges the outputs of the previous two activities and consults a loan expert who defines the approval decision, which is stored in a variable named *requestState*. The process is depicted in Figure 1 using BPMN notation. This example introduces process elements such as a split, a merge, a subprocess, a set of activities, and two decision gateways.

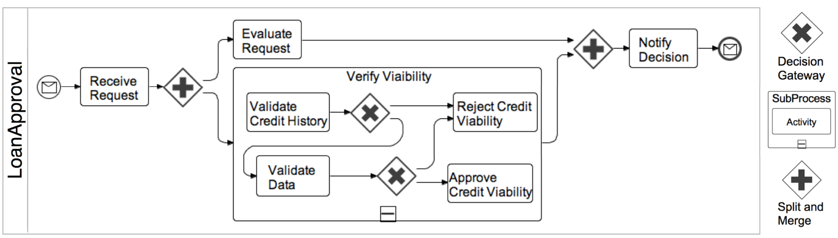


Fig. 1. Loan Request and Approval Process

This BPMN model can be translated into an executable process implementation (*e.g.,* BPEL), thus the process can be automated [11].

2.2 Business Process Analysis Requirements

Once the loan process is automated, the process analysts require feedback during process execution to enable its improvement. For example, based on feedback they could maximize the number of approved loan requests by adapting the business process constraints used by the *EvaluateRequest* activity. In order to support analysis activities, the following requirements are applicable:

1. *Monitoring events during process execution.* It is for example necessary to obtain the state of the loan request variable (*requestState*) for each loan process instance when the request is approved or rejected. This behavior occurs when the loan request variable is updated in the *NotifyDecision* activity.

2. *Creating new measurement data associated with process execution concepts and domain-specific concepts.* Measurement data is constructed by using process execution data as well as existing measurement data. For example, the process analysts require constructing new domain-specific metrics such as accepted/rejected requests, total requests, accepted/rejected requests ratio.
3. *Defining control rules using measurement data to understand process execution.* For example, the process analysts define a control rule to compute the defined measurement data and to verify when the rejected requests rate is higher than 50%, which means that the causes for rejection need to be verified (*e.g.,* the requested amount is too high). As soon as this behavior is present in the process, a control action can be taken. For example by notifying this situation to the quality assurance manager to allow for manually taking the corrective actions to improve the process (*e.g.,* adapt a business to define that the requested amount to automatically approve a request is $4500, instead of $5000) .

3 The MMC-BPM Language

MMC-BPM is a Domain Specific Language to complement business process models with monitoring, measurement and control (MMC) concerns. The language offers a declarative specification in which analysts describe what the analysis concern does instead of describing how it is implemented.

An MMC model defines the analysis concern associated with a target process from a specific observation point defined by the analysts. For example, an administrative stakeholder can be interested in analyzing the process using metrics such as rejected loan rates, while a quality control stakeholder could be interested in analyzing the process using time-related metrics such as average processing time. An MMC specification contains three main blocks to group the language constructs: the *Data* block, the *Event* block, and the *MMCRule* block. The code in Figure 2 illustrates how to specify the requirements presented in the case study using our MMC-BPM language. This specification corresponds to a specific observation point of a loan expert (line 1). We refer to this analysis specification throughout the section to illustrate our language elements.

3.1 Data Block

A process analyst uses the data block to define a) the data manipulated in the process and their types, and b) new measurement data required to analyze the process (Figure 3).

The MMC-BPM language provides a set of constructs to describe the data associated with the process model and with the elements in it (*e.g.,* Activities). This data model facilitates the use of process information within the analysis specification. For example, an analyst can describe that the *requestState* variable is associated to the *NotifyDecision* activity. Note that the data model is specified externally to the MMC model. As a consequence it can be reused in

```
1  MMCspec QualityView process LoanApproval
2   //Data
3   import dataTypes LoanProcess.xsd;
4   include data LoanProcess.pdata;
5
6   int AR; int RR; int TR; double RRA; double RRR;
7   //Events
8   event requestStateChange parameters boolean request;
9
10  onChange data.NotifyDecision.requestState
11   generates requestStateChange using data.NotifyDecision.requestState;
12  //Rules
13  mmcrule UpdateRequestState onEvent requestStateChange do
14   TR = TR + 1;
15   if !(event.request) then
16    RR = RR + 1; RRR = RR / TR;
17   else
18    AR = AR + 1; RRA = AR / TR;
19   endif
20   if RRR*100 > 50 then
21    notify 'quality@bank.com' 'RRR is too high' 'review the log.';
22   endif
23  endRule
24 endMMC
```

Fig. 2. Monitoring, Measurement and Control Specification for the Loan Process

```
Data             ::= ProcessDataTypes ProcessData (MeasureConcept)*;
ProcessDataTypes::= "import" "dataTypes" dataTypesFile=ID ".xsd" ";";
ProcessData      ::= "include" "data" dataFile=ID ".pdata" ";" ;
MeasureConcept   ::= type=DataType measureName=ID ";";
DataType         ::= (SimpleType | ComplexType | CollectionType);
SimpleType       ::= ("string" | "boolean" | "int" | "float" | "double" );
ComplexType      ::= complexDataType=ID;
CollectionType   ::= "<" (simpleType=SimpleType | complexType=ComplexType )">";
```

Fig. 3. EBNF specification of the Data Block

multiple analysis specifications. The MMC-BPM language references the data model using the include data <dataFile> clause (Line 4 in Figure 2).

In addition to the data model there is also a data type model, which determines the data types associated with the process data. We assume that the process data type model is defined using an XML Schema language. For example, the schema representing process data types can contain a complex data definition named *Client* containing a sequence of elements such as name, identification, and gender with their associated primitive data types. The data type model is referenced by using the import dataTypes <dataTypesFile> clause (Line 3 in Figure 2).

Measurement data defined in this specification is general to all the process instances. A measurement concept (*MeasureConcept*) is defined describing the name of the measurement data (*measureName*) and its data type (*DataType*). The MMC-BPM language provides a set of primitive data types (*SimpleType*) such as: string, int, float, boolean and double. For example, line 6 in Figure 2 illustrates the definition of the *accepted request (AR)* measurement concept using an *int* data type. Since business processes contain more complex data structures (*e.g.,* Loan, Client), the MMC language facilitates associating complex data types (*ComplexType*) and collections (*CollectionType*) to the measurement data.

A complex data type is related to one of the structures defined in the process data types model. Our DSL supports the definition of generic metrics (*e.g.,* duration, count, and resource) that are useful for any kind of process analysis and domain-specific metrics (*e.g.,* total number of rejected loans) that are only useful in a particular business domain.

3.2 Event Block

The Event block represents elements of the process domain used in the definition of analysis rules (for the purpose of monitoring) and the specific points where these rules are connected to the process model (Figure 4). A *ProcessEvent* is used to define the moment when a rule must be triggered as a result of an expected behavior (*e.g.,* an activity started) in the process execution. The MMC-BPM language allows grouping multiple process events in a concept named *LogicEvent*. This defines the information required to execute a rule.

Logic events detach rules from events, which fosters reuse. Thus, multiple process events can be associated with the same rule. For example, a process analyst can define two process events to retrieve the start and end time of all process activities. Thus, creating a rule for each process event, the code for computing the total processing time will be scattered in the analysis specification. The logic event makes it possible to modularize the rule code, which avoids putting cross-cutting code in the analysis specification. This feature of the language separates what users want to do from how to do it.

```
Event                ::= (logicEvent=LogicEvent)+ (processEvent=ProcessEvent)+;
LogicEvent           ::= "event" eventName=ID "parameters" ParameterSet ";";
ParameterSet         ::= parameter=Parameter("," parameter=Parameter)*;
Parameter            ::= type=DataType parameterName=ID;
ProcessEvent         ::= executionEvent = (ControlFlowEvent|DataFlowEvent)
                          "generates" eventName=ID "using" ParameterValueSet ";";
ControlFlowEvent     ::= executionEvent=ControlExecutionEvent BPMNModelIdentifier;
ControlExecutionEvent::= ("onStart" | "onFinish")
BPMNModelIdentifier  ::= "process""."bpmnElementName=ID;
DataFlowEvent        ::= executionEvent=DataExecutionEvent DataModelIdentifier;
DataExecutionEvent   ::= ("onChange" | "onMessage" | "onAlarm");
DataModelIdentifier  ::= "data"["."bpmnElementName=ID]"."attributeName=ID;
ParameterValueSet    ::= ParameterInvocation(","ParameterInvocation)*;
ParameterInvocation  ::= (DataModelIdentifier | MonitorInvocation | Literal |
                           SystemInvocation | EventInvocation);
MonitorInvocation    ::= "MonitorModel""."measureName=ID [InvocationType
                           parameterValueSet=ParameterValueSet];
InvocationType       ::= (invocationType="setValue"| invocationType="getValue");
SystemInvocation     ::= (service="TimeSystem");
```

Fig. 4. EBNF specification of the Event Block

Event blocks must contain at least one logic event and at least one process event. A logic event is defined using the `event <eventName> parameters <ParameterSet>` clause. The *eventName* identifies the event that will be used by a rule. The *ParameterSet* holds the information the logic event requires. Each *Parameter* contains a name and its corresponding *DataType*. Line 8 in Figure 2 illustrates a logic event named requestStateChange, which has a boolean parameter called request.

A process event describes a specific point in the process execution where it is necessary to a) capture process information, and b) generate a logic event assigning the retrieved information to the logic event parameters. This is done in the MMC-BPM language using the `<executionEvent> generates <eventName> using <ParameterValueSet>` clause. Figure 2 (lines 10-11) illustrates a process event that generates the logic event mentioned above when the variable *requestState* (associated with the activity *NotifyDecision*) is updated. The parameter of the logic event takes the value of this variable.

a) Capture Process Information. The MMC-BPM language offers the concept of an *executionEvent* to describe a relevant process execution behavior that we want to be aware of (*i.e.,* onStart, onFinish, onChange, onMessage, onAlarm). These execution events are defined in terms of process domain concepts without any specific implementation dependency. Our MMC-BPM language uses a subset of the vocabulary provided by the process domain model (*BPMNModelIdentifier*) or the process data model (*DataModelIdentifier*) to specify elements that the execution events monitor (*i.e.,* Activity, Data, IntermediateMessage). Process elements can be referenced using the `process.<bpmnElementName>` clause, where the *bpmnElementName* element can reference a unique process element (*e.g.,* NotifyDecision). Process data elements can be referenced using the `data.[<bpmnElementName>].<attributeName>` clause, where the *attributeName* references an attribute associated with a process element. Although we describe only some of the possible process elements and execution events available, the monitoring specification can be easily extended.

b) Generate a Logic Event. Once the process information is captured, a process event indicates which logic event must be generated (*eventName*). The parameter values (*ParameterValueSet*) of the logic event are assigned indicating a set of parameter invocations (*ParameterInvocation*). These parameter invocations can assign a value directly (*e.g.,* number, string), retrieve information from the process data model (*DataModelIdentifier*) or the monitoring data model (*MonitorInvocation*), or capture information from a system service (*SystemInvocation*). A monitor invocation facilitates accessing measurement data stored in the MMC data model. This is done using the `MonitorModel.<measureName>` clause with an optional set of parameter invocations (*ParameterValueSet*).

3.3 Rule Block

The rule block enables analysts to define actions to measure the process and to control its execution.

In our language, the `mmcrule <ruleName> onEvent <eventName> do` clause allows specifying the name of the rule and the logic event that will trigger the rule. The body of a rule can contain a) a set of actions, and b) a set of condition-action statements (Figure 5). Figure 2 (lines 13-23) illustrates the rule code for the case study.

```
MMCRule          ::= "mmcrule" ruleName=ID "onEvent" eventName=ID "do"
                       actionSet=ActionSet (conditionActions=ConditionAction)*
                     "endRule";
ActionSet        ::= (actionStatement=ActionStatement)*;
ActionStatement  ::= Assignment | ManagementAction;
Assignment       ::= measureName=ID "=" domainExpression=DomainExpression ";";
DomainExpression::= invocation+=Invocation(Oper invocation+=Invocation)*;
Invocation       ::= (ParameterInvocation | EventInvocation);
EventInvocation  ::= "event""." parameterName=ID;
ManagementAction::= NotifyAction | StoreAction | TraceAction ;
NotifyAction     ::= "notify" destinatary=STRING subject=STRING content=STRING";";
StoreAction      ::= "store" monitorInvocation=MonitorInvocation ";";
TraceAction      ::= "trace" log=STRING path=STRING ";";
ConditionAction  ::= "if" ifStatement=ConditionSet "then" thenStatement=ActionSet
                      ["else" thenStatement=ActionSet] "endif";
```

Fig. 5. EBNF Specification of the Rule Block

a) Actions. The *ActionSet* denotes a combination of action statements that can invoke an assignment (*Assignment*) to compute measurement data or take a management action (*ManagementAction*) to improve the process.

The measurement data is built using the `<measureName> = <Domain Expression>` clause. The *measureName* corresponds to a measurement data previously defined in the data block. The *DomainExpression* represents the arithmetical relation between a *ParameterInvocation* set or an *EventInvocation* set. The `event.<parameterName>` clause defines an event invocation. The event extracts the information required by the rule from a logic event definition. This measurement data facilitates process analysis in the particular business domain since these specifications are data-centric. The process execution data and measurement data are made persistent in the MMC data model for further analysis. Figure 2 illustrates the assignments to the measurement data using metrics stored in the MMC data model and other domain-specific metrics previously defined.

Process analysts can define management actions such as 1) storing monitoring concepts in the MMC model for visualization or query purposes (*store*), 2) the creation of a Log with the information retrieved and constructed in the application of the rule (*trace*), and 3) the notification of certain events to other external entities as for example the notification via mail (*notify*). Line 21 in Figure 2 illustrates a management action code for the case study, which notifies the quality assurance area when the rejected requests rate is higher than 50%.

b) Condition-Action statements. Process analysts can indicate the special behavior and the corresponding action to be taken in the Rule block to analyze the process. This is done with Condition-Action statements.

A control specification describes the *ConditionSet* that must be satisfied to execute some *ActionSet*. A set of conditions denotes a logic combination of expressions (*DomainExpression*). These represent an arithmetical relation between *ParameterInvocation* set or an *EventInvocation* set. The set of expressions involve a boolean expression, which triggers the set of management actions accordingly.

The language supports process analysis using evaluation statements over 1) the measurement data, 2) the process execution information, and 3) the process data. For example, Figure 2 (lines 15, 17, 20) illustrates the evaluation statements to verify the state of the loan request and the rejected requests rate percentage to apply some actions. Other examples, for analyzing the process, involve the classification of generic Service Level Agreement (SLA) metrics [9]. A business SLA can state that loan requests with an amount over a certain threshold must be rejected, and an IT SLA can state that 95% of the loan request must complete within 72 hours.

4 Integrating MMC Concerns with the Business Process

This section briefly presents our approach to transform the MMC model into an executable implementation. We transform the MMC model into (BPEL) aspects, which are later merged with the BPEL base process [10]. We base our approach on Model-Driven Engineering (MDE) technology [12].

4.1 Padus: An Aspect-Oriented Workflow Language

We use Padus, an aspect-oriented extension to BPEL, which allows introducing crosscutting behaviour to an existing BPEL process in a modularized way [13].

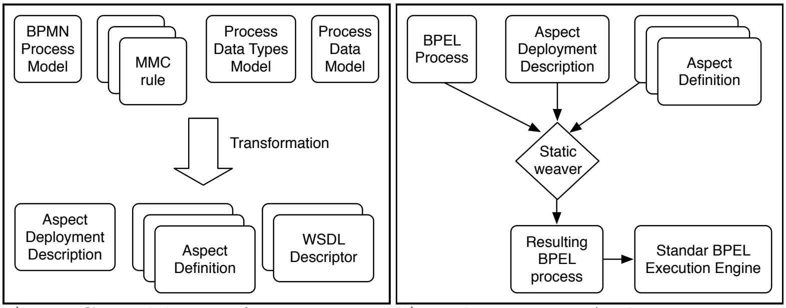
The Padus language facilitates the definition of specific points during the process execution where additional behaviour has to be added. These points can be selected using a logic pointcut language, and the Padus weaver can be used to combine the behaviour of the core process with the specified behaviour. The new behaviour can be introduced by inserting it *before* or *after* certain joinpoints defined by the pointcut, or it can replace existing behaviour by using an *around* advice. The Padus language introduces the concept of *in* advice to add new behaviour to existing process elements and provides an explicit deployment construct to specify aspect instantiation in specific processes. The advice code contains the extra behaviour that should be inserted, which is specified using standard BPEL elements.

4.2 Encapsulating Executable MMC Rules

Figure 6a illustrates the transformation process from the high-level analysis specification to the executable implementation.

The input for the transformation is a) the analysis specification defining the MMC rules, b) the BPMN process model representing the domain elements that we want to refer in our language, c) the XML schema describing the data types model associated with the process, and d) the process data model that describes the data attributes associated with each process element.

The output of the transformation is a) the Padus aspects that represent the MMC crosscutting concerns, b) the aspect deployment specification indicating how the aspects are instantiated and composed to the base process, c) the weaver java class that composes the base process and the aspects, d) the web services for supporting management actions, accessing the monitoring data model, or accessing to system services, and e) the XML schema that describes the measurement data defined in the language.



a) MMC Model Transformation b) Padus Weaver Architecture

Fig. 6. Transformation Process for the High-level Analysis Specification

Table 1. Analysis Specification into an Aspect Implementation

MMC-BPM language construct	*Output : Padus Implementation*
MMC model	- Aspect defining the analysis specification - Reference to the process model (BPEL code)
Data	Reference to data definitions (BPEL code)
LogicEvent	Joinpoint
ProcessEvent	Pointcut
onStart	Before advice
onFinish	After advice
onChange, onMessage	In advice
MMCRule	Advice (BPEL code)

Table 1 details the relevant mappings from our analysis language to the aspect-oriented programming implementation elements.

Figure 6b illustrates the Padus weaver architecture. The resulting artefact, after applying the aspect deployment description, is a BPEL process that can be deployed on a BPEL execution engine. We created an Eclipse plugin that serves as an editor to define the MMC rules and to execute the transformation process. The proposed MMC-BPM language and the model transformations rely on the OpenArchitectureWare [14] environment as metamodeling framework.

5 Related Work

Several commercial BPM products offer solutions for business activity monitoring (BAM) [15] [3] [4]. Typically, BAM solutions extract the information from audit trails, in which process metrics are added to the process architectures for analysis. In contrast, we propose a high-level language to describe, from a business perspective and independently of specific implementation languages, the process activities we want to monitor, the process and domain-specific metrics we want to build, and rules we want to apply over this information.

Other approaches use data mining and data warehouse techniques to capture the process execution information and to discover structural patterns for identifying the characteristics (explanations) that contribute to determine the value of a metric [16]. Our DSL allows end users to define evaluation statements making explicit the behavior that they are interested in and providing an implicit explanation of such situation. Moreover, our DSL allows defining the relations between monitoring and process data to built new domain-specific measures.

Other works [2] [5] [16] [17] introduce the idea to perform business process management using taxonomies and ontologies to capture semantic aspects. The authors of these works, as we do, consider analysis capabilities at the knowledge level, the definition of domain-specific metrics, and subsumption relations between concepts. The main difference is that our approach is focused on the analysis of business process activities instead of using process mining techniques. Process Mining aims at automatically discovering analysis information about processes based on event logs. Our approach facilitates to perform a similar performance analysis (measures) to the ones using a mining approach. Our approach can also be used to support the process mining approaches specializing the management action *trace* to provide the facilities necessary for logging the process and monitoring trail data in a structured way.

6 Conclusions and Future Work

Our approach helps in making the measurement variables and the associated rules more transparent to the end user by explicitly modeling them with a domain specific language and within an integrated process data model. In contrast to other approaches, domain-related metrics are easier to support because of the data model that we use to describe data associated with the process elements and the capabilities to define new measurement data. This high-level of abstraction facilitates the transformation of the analysis specifications into multiple workflow implementations. Our aspect-oriented approach in the definition and implementation of the analysis concerns offers process analysts the opportunity to evolve them independently of the process implementation.

Using our MMC-BPM language increases the level of abstraction of the business process analysis specifications at a conceptual level. In the future we plan to advance our work in several areas. Firstly, we require a way to assess how the MMC rules interfere with the process being executed. Secondly, it is necessary to find a mechanism to instrument the process for supporting the analysis specification done in terms of process data events. Third, we require defining a mechanisms to tackle or at least manage co-evolution issues between of MMC rules and the BPMN process model they refer to. Finally, as we consider that multiple users can analyze the same process, we should investigate ways to detect and resolve possible inconsistencies and conflicts that may arise.

References

1. van der Aalst, W.M.P., ter Hofstede, A.H.M., Weske, M.: Business process management: A survey. In: van der Aalst, W.M.P., ter Hofstede, A.H.M., Weske, M. (eds.) BPM 2003. LNCS, vol. 2678, pp. 1–12. Springer, Heidelberg (2003)
2. Hepp, M., Leymann, F., Domingue, J., Wahler, A., Fensel, D.: Semantic business process management: A vision towards using semantic web services for business process management. In: Lau, F.C.M., Lei, H., Meng, X., Wang, M. (eds.) ICEBE, pp. 535–540. IEEE Computer Society, Los Alamitos (2005)
3. Miers, D., Harmon, P., Hall, C.: The 2007 bpm suites report – a detailed analysis of bpm suites version 2.1, Business Process Trends (2007), http://www.bptrends.com/
4. Lau, C., Peddle, S., Yang, S.: Gathering monitoring metrics to analyze your business process, IBM (December 2007), http://www.ibm.com/us/
5. de Medeiros, A.K.A., Pedrinaci, C., van der Aalst, W.M.P., Domingue, J., Song, M., Rozinat, A., Norton, B., Cabral, L.: An outlook on semantic business process mining and monitoring. In: Meersman, R., Tari, Z., Herrero, P. (eds.) OTM-WS 2007, Part II. LNCS, vol. 4806, pp. 1244–1255. Springer, Heidelberg (2007)
6. OMG: Business process modeling notation specification - final adopted specification (February 2006), http://www.bpmn.org
7. IBM: Business process execution language for web services (July 2002), http://www.ibm.com/developerworks/library/specification/ws-bpel/
8. Dubray, J.J.: The seven fallacies of business process execution, InfoQ (December 2004), http://www.infoq.com/
9. Castellanos, M., Casati, F., Dayal, U., Shan, M.C.: Intelligent management of slas for composite web services. In: Bianchi-Berthouze, N. (ed.) DNIS 2003. LNCS, vol. 2822, pp. 158–171. Springer, Heidelberg (2003)
10. González, O., Casallas, R., Deridder, D.: Modularizing monitoring rules in business processes models. In: Meersman, R., Tari, Z., Herrero, P. (eds.) OTM Workshops. LNCS, vol. 5333, pp. 22–23. Springer, Heidelberg (2008)
11. Giner, P., Torres, V., Pelechano, V.: Bridging the gap between bpmn and ws-bpel. m2m transformations in practice. In: Koch, N., Vallecillo, A., Houben, G.J. (eds.) MDWE. CEUR Workshop Proceedings, vol. 261, CEUR-WS.org (2007)
12. Schmidt, D.C.: Guest editor's introduction: Model-driven engineering. IEEE Computer 39(2), 25–31 (2006)
13. Braem, M., Verlaenen, K., Joncheere, N., Vanderperren, W., Straeten, R.V.D., Truyen, E., Joosen, W., Jonckers, V.: Isolating process-level concerns using padus. In: Dustdar, S., Fiadeiro, J.L., Sheth, A.P. (eds.) BPM 2006. LNCS, vol. 4102, pp. 113–128. Springer, Heidelberg (2006)
14. OpenArchitectureWare (oAW) website, http://www.openarchitectureware.org
15. von den Driesch, M., Blickle, T.: Operational, Tool-Supported Corporate Performance Management with the ARIS Process Performance Manager. Aris in practice edn. (2006)
16. Grigori, D., Casati, F., Castellanos, M., Dayal, U., Sayal, M., Shan, M.C.: Business process intelligence. Computers in Industry 16(3), 321–343 (2004)
17. Pedrinaci, C., Domingue, J., Brelage, C., van Lessen, T., Karastoyanova, D., Leymann, F.: Semantic business process management: Scaling up the management of business processes. In: ICSC, pp. 546–553. IEEE Computer Society, Los Alamitos (2008)

Strategy-Driven Business Process Analysis

Carlos Pedrinaci[1], Ivan Markovic[2], Florian Hasibether[2], and John Domingue[1]

[1] Knowledge Media Institute, The Open University, Milton Keynes, MK7 6AA, UK
{c.pedrinaci,j.b.domingue}@open.ac.uk
[2] SAP Research, Karlsruhe, Germany
{ivan.markovic,florian.hasibether}@sap.com

Abstract. Business Process Analysis (BPA) aims to verify, validate, and identify potential improvements for business processes. Despite the wide range of technologies developed so far, the large amount of information that needs to be integrated and processed, as well as the quantity of data that has to be produced and presented still poses important challenges both from a processing and presentation perspectives. We argue that to enhance BPA, semantics have to be the core backbone in order to better support the application of analysis techniques on the first hand, and to guide the computation and presentation of the results on the other hand. We propose a knowledge-based approach to supporting strategy-driven BPA by making use of a comprehensive and extensible ontological framework capturing from high-level strategic concerns down to lower-level monitoring information. We describe how corporate strategies can be operationalized into concrete analysis that can guide the evaluation of organisational processes, structure the presentation of results obtained and better help assess the well-being of corporate business processes.

Keywords: Semantic Business Process Management, Business Process Analysis, Strategic Analysis.

1 Introduction

In the business world the maxim "*if you can't measure it, you can't manage it*" is often used. Although it is too blunt a statement, it captures an important essence in current management approaches which try to maximise the aspects measured in order to evaluate, compare, and control the evolution of businesses. For instance the Balanced Scorecard is a popular "*set of measures that gives top managers a fast but comprehensive view of the business*" [1]. In a nutshell, the Balanced Scorecard defines four perspectives and suggests for each of them a set of aspects that managers should focus on. Assessing how well a company is doing is then a matter of calculating metrics and contrasting them with respect to pre-established goals for each of these key aspects.

Analysing business processes in an effective manner requires computing metrics that can help determining the health of business activities and thus the whole enterprise. However, this is not all there needs to be done. Aspects like

W. Abramowicz (Ed.): BIS 2009, LNBIP 21, pp. 169–180, 2009.

client's satisfaction, whether a certain strategy will work out or not, how successful the research department is, or what would happen if we make a certain change in a process cannot "simply" be measured. Similarly, detecting, or better yet, anticipating process deviations with respect to expected behaviours can hardly be approached as a simple measurement problem. The closer we get to strategic analysis, the more impact analysis results are likely to have, but the more complex analysis techniques are required in order to deal with qualitative aspects, approximations, and uncertainty. In order to deal with these scenarios, BPA solutions need to apply advanced analysis techniques.

Semantic BPM, that is, the extension of Business Process Management (BPM) with Semantic Web and Semantic Web Services technologies has been proposed as a means for increasing the level of automation during the life-cycle of business processes [2]. This vision is pursued within the SUPER project[1] as part of a SBPM framework aiming at increasing the level of automation and enhancing the support for managing business processes by using comprehensive semantic models of the BPM domain. As part of this initiative, we advocate the use of a holistic integrated view of the enterprise spanning from high-level strategic concerns down to low-level operational details, in order to increase the level of automation in BPA and to better support the continuous improvement of strategic decision-making processes. In this paper, we focus on the conceptual models for capturing strategies and how they can be used for enhancing the analysis of business processes thanks to a formal and explicit modelling of the relations between strategies and operational aspects.

The remainder of the paper is organised as follows. We first introduce Core Ontology for Business Process Analysis (COBRA), which provides the core foundational conceptualisation for our work. Section 3 presents an extension of COBRA towards capturing different kinds of operational analyses that typically take place within BPM systems. Conversely, Section 4 describes Business Motivations Ontology (BMO) another extension of COBRA in this case focussed on strategic concerns and their operationalisation into outcomes that can be evaluated. Section 5 describes in more detail how our conceptual framework can support and guide the assessment of the well-being of enterprises by evaluating the desired outcomes that have strategic importance. Finally, we contrast our research with previous work in the area and we present our main conclusions and future work.

2 COBRA

Supporting the level of automation demanded by enterprises nowadays requires enhancing BPA with support for applying general purpose analysis techniques over specific domains in a way that allows analysts to use their particular terminology and existing knowledge about their domain. To this end we have defined Core Ontology for Business pRocess Analysis [3]. COBRA provides a core terminology for supporting BPA where analysts can map knowledge about some

[1] http://www.ip-super.org

particular domain of interest in order to carry out their analyses. It is worth noting that COBRA does not aim to provide a fully-comprehensive conceptualisation for supporting each and every kind of analysis since the scope would simply be too big to be tackled appropriately in one ontology. Instead COBRA, depicted in Figure 1, provides a pluggable framework based on the core conceptualisations required for supporting BPA and defines the appropriate hooks for further extensions in order to cope with the wide-range of aspects involved in analysing business processes.

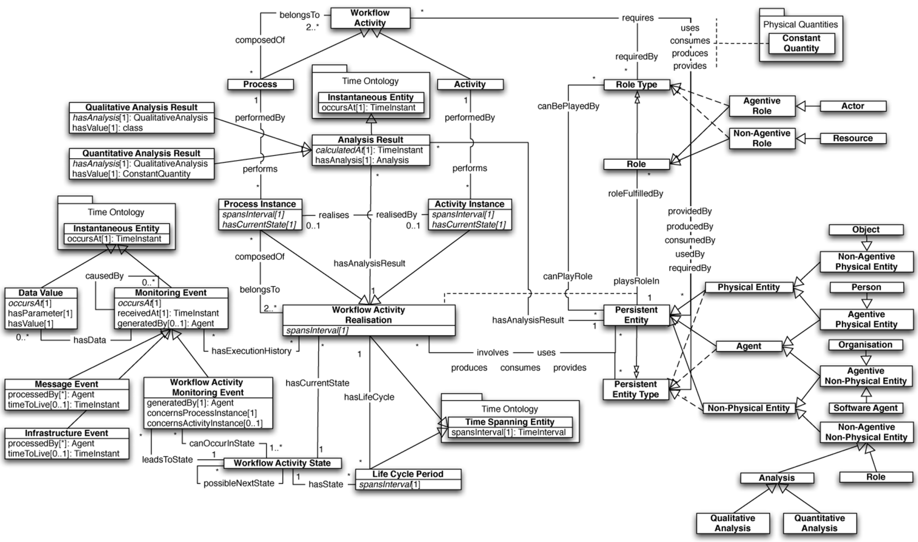


Fig. 1. Core Ontology for Business pRocess Analysis

COBRA builds upon Time Ontology that provides a temporal reference by means of which one can determine temporal relations between elements. COBRA provides a lightweight foundational basis in order to ensure a coherent view among additional extensions. It defines concepts such as *Process*, *Activity*, *Process Instance*, *Role*, *Resource*, *Organisation* or *Person* which are to be refined within specific ontologies as defined within SUPER, or other approaches like the Enterprise Ontology [4] or TOVE [5].

COBRA has been extended with a reference Events Ontology (EVO) [3] that provides a set of definitions suitable to capture monitoring logs from a large variety of systems and ready to be integrated within our core ontology for analysing business processes. It is based on existing syntactic formats, e.g., MXML [6] or the Audit Trail Format by the Workflow Management Coalition [7] which therefore confers on it the ability to capture logs generated by a plethora of systems. As prescribed by COBRA, EVO is centred around a state model that accounts for the status of processes and activities. The state model has been captured ontologically and enhanced with additional relations, , see [3] for further details.

Concerning the analyses themselves such as metrics, the previously presented version of COBRA [3] solely captures the concept *Analysis Result*, a Temporal Entity, which has two disjoint sub-concepts: *Qualitative Analysis Result* and *Quantitative Analysis Result*. As part of our work on strategy-driven analysis as well as metrics definition and computation, we have slightly extended COBRA itself. First, COBRA includes support for units of measure and their manipulation. Secondly, we have introduced the concept *Analysis*, which is refined into *Qualitative Analysis* (e.g., "is the process critical?") and *Quantitative Analysis* (e.g., "process execution time") based on the type of Analysis Result they produce. This provides us the means for maintaining a library of Analysis specifications (e.g., metrics, time series, etc.), and it allows us to distinguish between the analysis themselves and the actual results. Indeed, the relationship between Analysis and Analysis Result has also been captured, in such a way that every Analysis Result is a result for a particular Analysis, and every Analysis may have several Analysis Results. Hence we can obtain all the results for a particular analysis, track its evolution over time, apply time series analysis, etc.

3 Operational Analysis Ontology

Operational Analysis Ontology is an extension of COBRA focussed on supporting operational analysis, that is the analysis over operational aspects that can directly be computed. More abstract concerns such as strategic analysis being captured in another ontology that will be described in Section 4.

Operational Analysis Ontology, depicted in Figure 2, is mainly based on the notions of Analysis and Analysis Result defined in COBRA. In particular it defines the main kinds of analyses over operational aspects and their respective results. Central to Operational Analysis Ontology is therefore the concept of Metric which supports the definition of operational measures used for monitoring the well-being of business processes and enterprises. The concrete definition of Metric as well as the means for computing them will be explained in more detail in Section 3.1. Additionally the ontology defines another Quantitative Analysis, namely *Quantified Quantitative Analysis Variation*. This kind of analysis basically captures the variation of the value of a certain Quantitative Analysis over a certain period (e.g., "the Process Execution Time increased by 5 ms"). Similarly, we also capture the variation qualitatively as *Quantitative Analysis Variation* for which the results are solely the *Variation Type*, e.g., *Increased*, *Strictly Decreased*.

Finally, the ontology captures two Qualitative Analyses, namely *Qualitative Analysis Comparison* and *Quantitative Analysis Comparison*. The former allows us to specify things like "is the Process a bottleneck?" whereby determining the kind of Process is a Qualitative Analysis and "bottleneck" is one of the kinds of results that can be obtained. Defining these kinds of Qualitative Analysis Comparison is based on the specification of a Qualitative Analysis, a *Logical Comparison*, and the value to compare to. Among the Logical Comparisons contemplated we include things like equality or subsumption relations. Conversely,

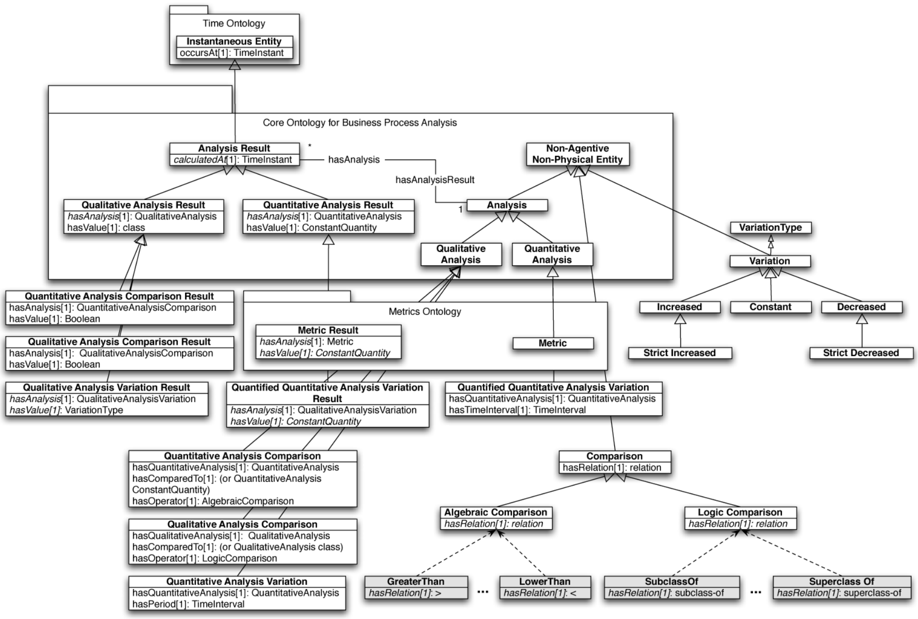


Fig. 2. Operational Analysis Ontology

Quantitative Analysis Comparison supports the definition of comparison between any Quantitative Analysis (e.g, a particular Metric) and a reference value which could itself be another Quantitative Analysis. In this way one can define things like "is the Process Execution Time > 10 ms?" or "is Process A Execution Time > Process B Execution Time?".

3.1 Metrics Ontology

Operational Analysis Ontology has been defined in rather abstract terms without explaining the mechanisms by which one could take these definitions and apply within a concrete domain. In this section we focus on Metrics Ontology which allows us to define metrics in way that can support their automated computation. The reader is referred to [8] for the details of SENTINEL, a monitoring tool that makes use of Metrics Ontology for computing and presenting metrics about the execution of business processes.

Metrics Ontology provides us with the capacity for specifying and computing metrics, as necessary for analysing and managing business processes, in a domain-independent way. On the basis of our conceptualisation we can capture kinds of metrics, e.g,. "process instance execution time", as well as specific metrics to be computed, e.g., "process instance X execution time". The former are defined as concepts, whereas the latter are modelled as instances. In this way we can provide libraries of metrics such as general purpose ones, or specific for some domain like Supply-Chain, and at analysis time the analyst can specify which of these

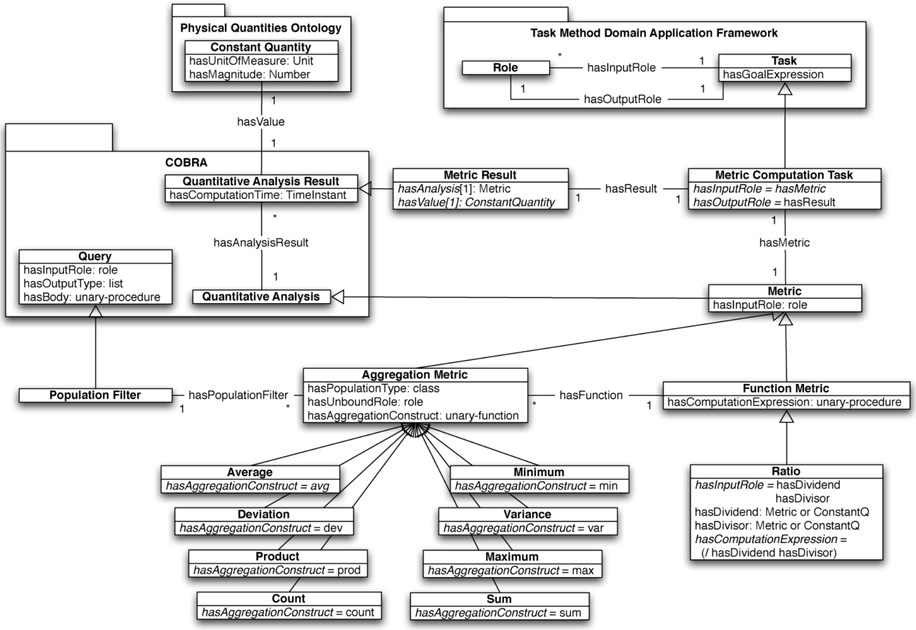


Fig. 3. Metrics Ontology

metrics should be computed over which entities by instantiating them. This provides a convenient way for organising metric definitions and seamlessly supports the comparison of results by kind of metric, e.g., "which is the process which takes longer", as well as it allows tracking their evolution over time.

Central to Metrics Ontology is the concept *Metric* which is defined as a Quantitative Analysis (see Section 2). Metrics are specified by a set of input roles that point to domain-specific knowledge [9]. We refine Metrics into two disjoint kinds, Function Metrics and Aggregation Metrics. A Function Metric is a metric that can be evaluated over a fixed number of inputs. For example, the Metric *Process Instance Execution Time* is a Function Metric which takes as input one Process Instance. Conversely, Aggregation Metrics (e.g., "average process execution time") take an arbitrary number of individuals of the same kind as input (e.g., a set of Process Instances). Therefore, Aggregation Metrics are computed over a population in order to obtain an overall perception of some aspect of interest such as the average execution time of some particular process. The population to be processed can be defined intensionally as an ontological query so that the metric computation can focus on certain processes, or resources of interest. In this respect the use of semantic technologies plays a key role towards supporting business analysts in the analysis of processes, allowing them to use their domain-specific terminology and still use a generic machinery to process the information in a seamless way.

In order to support the automated computation of metrics, which is indeed metric dependent, each metric has a computation expression which is defined as a unary procedure. In this respect it is worth noting that the language used to define the metrics themselves as well as to develop the metrics computation engine is Operational Conceptual Modelling Language (OCML) [9]. OCML seamlessly supports the integration of static and dynamic knowledge paving the way for a rapid prototyping of a fully operational solution[2].

4 Business Motivation Ontology

The purpose of the Business Motivation Ontology (BMO) is to allow for representation of the notions important for the strategic aspect of business process analysis. BMO is inspired by the Business Motivation Model [10], a business modeling standard published by the Object Management Group (OMG). BMO is a result of conceptual reengineering, refinement and formalization of concepts described in the standard specification. There are four top level concepts that are core to the BMO: Means, End, Metric and Influencer. We visualize BMO concepts and relations in Figure 4, where the imported concepts from COBRA are marked with gray boxes.

Ends refer to any aspiration concept (cf. Figure 4, upper right). They state, *what* an enterprise wants to be. This could be about changing what the enterprise is, as well as maintaining the actual position. Ends subsume the concepts Vision (abstract End) and Desired Outcome (concrete End). A Vision describes a future, ultimate, possibly unattainable state, an enterprise as a whole wants to achieve. It is made operative by Mission and amplified by Goals. A Desired Outcome is an attainable state or target that the enterprise, or some part of it, intends to achieve. Desired Outcomes are established for a certain time interval spanning the period of time for which the Desired Outcome holds. We distinguish between outcomes that need to be achieved at the end of the period, and those that have a periodic check by means of which one can define what we refer to as *continuous* Desired Outcomes. Continuous Desired Outcomes have the particularity that they specify outcomes that are continuously desired during a given interval and that will be checked periodically. This allows to express things like "sales should increase by 1 percent per month for the next year" as well as "increase sales by 5 percent by the end of the year" in a simple and concise way.

Given the high-level of abstraction Desired Outcomes can have, it is particularly difficult to assess the level of achievement currently attained, and decide how to map these desires into concrete enterprise-wide actions and decisions. In consequence, we support the decomposition of Desired Outcomes into finer grain ones, as a gradual process that takes us from purely strategic outcomes to the operational level where one can perform measures and contrast the achieved results with those desired. This gradual refinement of Desired Outcomes is supported through the refinement of Desired Outcomes into *Goals* and *Objectives*, and by using what we refer to as *Logical Decomposition* constructs. Goals tend to be longer term, qualitative, and ongoing in comparison to Objectives. An

[2] The ontologies described herein can be found at http://www.cpedrinaci.net

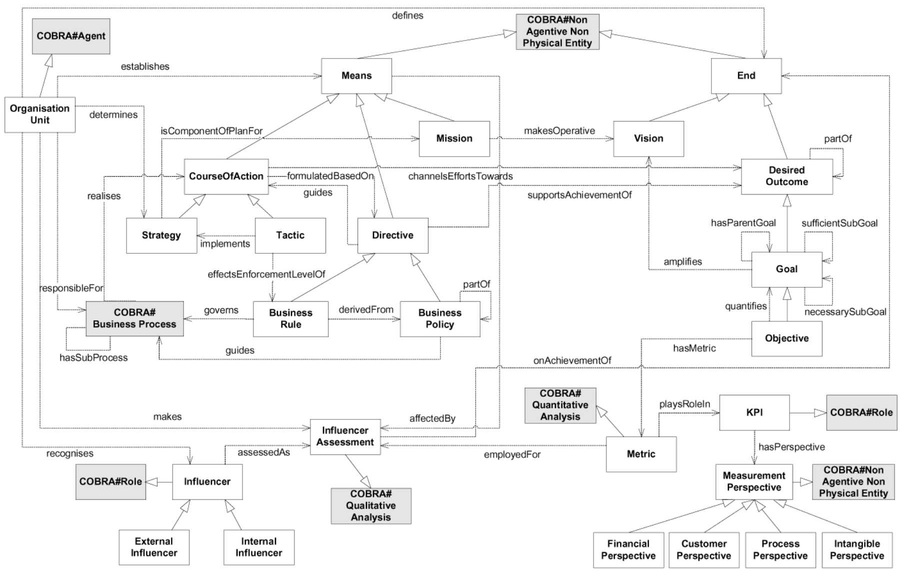


Fig. 4. Business Motivation Ontology

Objective is a measurable, time-targeted Goal. It is a step along the way towards achieving a Goal.

Means (cf. Figure 4, upper left) are counterparts of Ends - they state *how* an enterprise intends to achieve its Ends. A Means may be either a Mission, a Course of Action or a Directive. A Mission describes a long-term approach which is focused on achieving the Vision. Its definition should be broad enough to cover all possible strategies and the complete area of enterprise operations. A Course of Action is an approach (plan) for configuring some aspect of the enterprise, undertaken to achieve its Desired Outcomes. It includes the subconcepts Strategy and Tactic. A Strategy is accepted by the enterprise as the right approach to achieve its Goals. Each Strategy is implemented by Tactics, which tend to be shorter term and narrower in scope in comparison to Strategies. A Strategy may be implemented by several Tactics and a Tactic may contribute to the implementation of more than one Strategy. To clarify the distinction, we say Strategies channel efforts towards Goals and Tactics channel efforts towards Objectives. Thus, the two concepts establish an analogy for concepts on different level of abstraction on the complementary sides of Ends (the *what*) and Means (the *how*). A Course of Action is realized by Business Processes.

Directives are set up to guide the Courses of Action. A Directive defines or constrains some aspect of an enterprise. A Business Policy is a non-actionable Directive whose purpose is to guide or govern the Business Processes within an enterprise. It is not focused on a single aspect of governance or guidance. Business Policies provide the basis for Business Rules. A Business Rule is a Directive that is directly actionable. It is often derived from (implements) Business Policy.

Business Rules govern Business Processes and their enforcement level is effected by chosen Tactics.

Besides the fact that Metrics play an important role in business process analysis, setting targets is crucial to the motivational perspective of an enterprise's process space. BMO follows the Balanced Scorecard approach [1] here, and assigns performance measures with the company's aspirations. A Metric takes the role of a Key Performance Indicator (KPI) for an Objective, if it is applied to indicate the performance for a particular Objective. A KPI can take several perspectives; Financial, Customer, Process and Intangible (sometimes called Innovation and Learning).

An Influencer is something that can cause changes that affect the enterprise in the employment of its Means or achievement of its Ends. Almost anything within or outside a company could act as an Influencer in some situation. An Influencer Assessment is the procedure of judging the impact of an Influencer on an enterprise. Various methods can be used for performing this assessment. Consider an example method of SWOT analysis [11], where the result of Influencer Assessment can fall in the following categories: Strength, Weakness, Opportunity and Threat.

5 Strategy-Driven Analysis

The *what* vs. *how* duality of BMO allows us to contrast the strategies implemented with respect to established outcomes, i.e., "is the strategy contributing adequately to our ends?" Furthermore, it supports the analysis phase by guiding the computation and presentation of information based on what are the most important outcomes that need to be assessed. Evaluating the strategies themselves and determining whether the established Ends and their decomposition were reasonable, despite being very interesting research topics, fall outside of the scope of this paper.

As we indicated previously, Objectives are low-level Desired Outcomes that specify a desired condition over directly measurable aspects. These conditions are algebraic comparisons (e.g., greater than)[3] between any two Quantitative Analyses or between a Quantitative Analysis and a Constant Quantity (e.g., 5 ms). The specification of these conditions follows the approach established by COBRA in the sense that it distinguishes the analysis specification from the actual results obtained at certain points on time (e.g., see Quantitative Analysis Variation and Quantitative Analysis Variation Result). Objectives therefore allow us to define conditions over Metrics or their evolution over time (see Quantitative Analysis Variation) which can be computed directly from monitoring data (e.g., "execution time $>$ 10 days"). This allows us to bring the results to a higher-level of abstraction where one can talk about Objectives in terms of whether they have been met or not.

Conversely, Goals are qualitative Desired Outcomes defined at a higher-level of abstraction, thus more closely related to the final strategical outcomes wanted.

[3] Figures do not include all the operators for the sake of clarity.

Goals can be decomposed into subgoals by using Logical Decompositions. We currently contemplate two kinds of Logical Decompositions namely *conjunctions* and *disjunctions*. Conjunctions allow us to express that a certain Goal is achieved when all its subgoals are achieved. Disjunctions on the other hand specify that a Goal is achieved when at least one of its subgoals has been achieved.

To support the definition of Goal graphs of an arbitrary complexity and still be able to determine their assessability, we distinguish two kinds of Goals: *Operational Goals* and *Complex Goals*. Operational Goals are those that are defined as a composition of Objectives or Qualitative Analysis Comparisons (e.g., “is the QoS increasing?”). Operational Goals are therefore low-level Goals which can be directly assessed based on existing measures and by performing the appropriate metrics computation and/or comparisons. Complex Goals on the other hand are composed of Operational Goals and therefore support defining higher-level Goals while still retaining their assessability given that they are defined in terms of Goals which are directly assessable. The notion of assessability introduced above allows us distinguish Goals that can be evaluated automatically–*Assessable Desired Outcomes*–from those that can not–*Non Assessable Desired Outcomes*. Whereby Assessable Desired Outcomes are either Objectives, Operational Goals, Complex Goals, or Goals solely composed of assessable Goals. Non Assessable Desired Outcomes are those that do not meet the restrictions above.

Assessing Desired Outcomes is envisaged as a process like the one formalised in [12]. In their paper, Giorgini et al. describe both a qualitative and a quantitative model for evaluating goal models defined as AND/OR decompositions. Their model includes the formalisation of a set of propagation rules and a label propagation algorithm which is sound, complete, and scalable. They also include an extensive set of relations between goals that can capture typical situations such as the fact that Goals contribute or hinder each other, or even mutually impede or ensure the achievement of each other. The reader is referred to [12] for concrete details of their approach. The assessment of Desired Outcomes in our model can straightforwardly be implemented in the same way. More importantly, our concept of Objective being directly linked to measurable aspects, can be used as what the Giorgini et al. refer to as initial nodes and allow us to directly apply the evaluation of goals over monitored data without the need to introduce any data manually. As a consequence, our conceptual model i) supports the operationalisation of desired outcomes into a graph that can automatically be evaluated; and ii) allows for directing the analysis of processes and organisations based on those aspects which are known to be of strategic importance.

6 Related Work

With Balanced Scorecard [1], Kaplan and Norton devised an approach to strategy implementation based on the specification of strategic objectives and assignment of respective measures used for assessing the achievement of these objectives, according to the Scorecard perspectives. In contrast to our work, the Balanced Scorecard does not provide an explicit linkage between the strategic and operational aspects of a process-centric enterprise neither does it specify the evaluation mechanisms necessary to automate the assessment of objectives within the Scorecard.

In more general work on BPA, we find techniques based on plain statistical analysis, data and process mining techniques, neural networks, case-based reasoning, or heuristic rules [13, 7, 14–18]. Some techniques focus on automating analysis to the greatest extent whereas others pay particular attention to obtaining results that can easily be explained and presented to the user. What can be distilled is that researchers have so far focused on operational aspects, leaving strategic concerns up to analysts interpretation. The work described herein is therefore, to the best of our knowledge, the first attempt that provides a comprehensive conceptualisation ranging from strategic aspects to operational details in a way that is amenable to performing automated analysis driven by high-level strategic concerns.

7 Conclusions

Current practices in Business Process Management try to maximise what can be measured in order to better support the decision-making process. We have argued that in order to reach the level of automation demanded by businesses nowadays, we need a holistic integrated conceptual model of the enterprise spanning from high-level strategic concerns, down to low-level operational concerns. We have described our model based on a set of modular ontologies which are currently being developed an enhanced within the SUPER project as part of a wider conceptualisation of the BPM domain. The ontologies we have defined are aligned with a formal model for evaluating Goals which supports the development of sound, complete and scalable algorithms for the evaluation of strategic outcomes out of low-level monitoring details. Our conceptual framework represents, to the best of our knowledge, the most comprehensive approach developed for supporting automated strategic analysis within BPM.

Despite the promising conceptual results obtained and the firm foundations upon which our work is based, a fully automated system based on these notions needs to be implemented and thoroughly tested within real settings. Additionally, the establishment of methodologies and techniques for mapping concrete ends into particular means and their ulterior validation when deployed in concrete settings are indeed very interesting areas for further research.

References

1. Kaplan, R.S., Norton, D.P.: The Balanced Scorecard - Measures that Drive Performance. Harvard Business Review (January/February 1992)
2. Hepp, M., Leymann, F., Domingue, J., Wahler, A., Fensel, D.: Semantic Business Process Management: A Vision Towards Using Semantic Web Services for Business Process Management. In: Lau, F.C.M., Lei, H., Meng, X., Wang, M. (eds.) ICEBE, pp. 535–540. IEEE Computer Society, Los Alamitos (2005)

3. Pedrinaci, C., Domingue, J., Alves de Medeiros, A.K.: A Core Ontology for Business Process Analysis. In: Bechhofer, S., Hauswirth, M., Hoffmann, J., Koubarakis, M. (eds.) ESWC 2008. LNCS, vol. 5021, pp. 49–64. Springer, Heidelberg (2008)
4. Uschold, M., King, M., Moralee, S., Zorgios, Y.: The Enterprise Ontology. Knowledge Engineering Review 13(1), 31–89 (1998)
5. Fox, M.S.: The TOVE Project Towards a Common-Sense Model of the Enterprise. In: Belli, F., Radermacher, F.J. (eds.) IEA/AIE 1992. LNCS, vol. 604, pp. 25–34. Springer, Heidelberg (1992)
6. van Dongen, B., Alves de Medeiros, A.K., Verbeek, H., Weijters, A., van der Aalst, W.: The ProM Framework: A New Era in Process Mining Tool Support. In: Ciardo, G., Darondeau, P. (eds.) ICATPN 2005. LNCS, vol. 3536, pp. 444–454. Springer, Heidelberg (2005)
7. Mühlen, M.: Workflow-based Process Controlling. Foundation, Design, and Implementation of Workflow-driven Process Information Systems. Advances in Information Systems and Management Science, vol. 6. Logos, Berlin (2004)
8. Pedrinaci, C., Lambert, D., Wetzstein, B., van Lessen, T., Cekov, L., Dimitrov, M.: SENTINEL: A Semantic Business Process Monitoring Tool. In: Ontology-supported Business Intelligence (OBI 2008) at 7th International Semantic Web Conference (ISWC 2008), Karlsruhe, Germany (2008)
9. Motta, E.: Reusable Components for Knowledge Modelling. Case Studies in Parametric Design Problem Solving. Frontiers in Artificial Intelligence and Applications, vol. 53. IOS Press, Amsterdam (1999)
10. Object Management Group: Business motivation model (2006), `http://www.omg.org/docs/dtc/06-08-03.pdf`
11. Ansoff, H.I.: Corporate Strategy. McGraw-Hill, New York (1965)
12. Giorgini, P., Mylopoulos, J., Nicchiarelli, E., Sebastiani, R.: Formal reasoning techniques for goal models. Journal of Data Semantics (2004)
13. Castellanos, M., Casati, F., Dayal, U., Shan, M.C.: A comprehensive and automated approach to intelligent business processes execution analysis. Distributed and Parallel Databases 16(3), 239–273 (2004)
14. Grigori, D., Casati, F., Castellanos, M., Dayal, U., Sayal, M., Shan, M.C.: Business Process Intelligence. Computers in Industry 53(3), 321–343 (2004)
15. Azvine, B., Cui, Z., Nauck, D.D., Majeed, B.: Real Time Business Intelligence for the Adaptive Enterprise. In: The 8th IEEE International Conference on E-Commerce Technology. IEEE Computer Society, Los Alamitos (2006)
16. Watson, H.J., Wixom, B.H.: The Current State of Business Intelligence. Computer 40(9), 96–99 (2007)
17. March, S.T., Hevner, A.R.: Integrated decision support systems: A data warehousing perspective. Decision Support Systems 43(3), 1031–1043 (2007)
18. Alves de Medeiros, A.K., Van der Aalst, W., Pedrinaci, C.: Semantic Process Mining Tools: Core Building Blocks. In: 16th European Conference on Information Systems, Galway, Ireland (June 2008)

Case Construction for Mining Supply Chain Processes

Kerstin Gerke[1], Jan Mendling[2], and Konstantin Tarmyshov[1]

[1] SAP Research, CEC Dresden
Chemnitzerstr. 48, 01187 Dresden, Germany
kerstin.gerke@sap.com, konstantin.tarmyshov@sap.com
[2] Humboldt-Universität zu Berlin
Unter den Linden 6, 10099 Berlin, Germany
jan.mendling@wiwi.hu-berlin.de

Abstract. Process mining algorithms aim at the automatic extraction of business process models from logs. Most of these algorithms perform well on single-system event traces that explicitly refer to a process instance or case. However, in many operational environments such case identifiers are not directly recorded for events. In supply chain processes there are even further challenges, since different identification numbers, vertical integration and numerous aggregation steps prevent individual work steps to become traceable as a case. As a result, there are little experiences with the use of process mining in supply chains. To address this problem, we consider Radio Frequency Identification (RFID) for identifying the movements of the business objects. Based on an example process from the Supply Chain Operations Reference Model (SCOR), we highlight the two key challenges of making RFID events available for process mining: case identification and focus shifts. We demonstrate how RFID events that conform to the EPCglobal standard can be used to construct cases such that process mining can be applied. A respective algorithm is sketched that we implemented in a tool which generates MXML process mining data from EPCglobal event traces. In this way, we provide a contribution towards applying process mining techniques for supply chain analysis.

Keywords: Process Mining, Supply Chain Management, RFID, EPC-global, Business Process Modeling.

1 Introduction

Emerging technologies for interorganizational information systems provide new opportunities to analyze and optimize supply chain processes globally as in the past this was only possible in a local, intraorganizational setting. A prominent example that highlights the importance of global optimization is the so-called bullwhip effect that can be observed in supply chains [1]: due to delay of information processing between different parties and information aggregation, it can happen that small demand variation with retailers can build up to excessive demand variation for early suppliers at the beginning of the supply chain.

W. Abramowicz (Ed.): BIS 2009, LNBIP 21, pp. 181–192, 2009.

Information sharing is one of the strategies to circumvent such problems as the bullwhip effect, and Radio-Frequency Identification (RFID) is discussed as a technology to achieve this information sharing [2].

While there is consensus about the need for global supply chain analysis, there has been little work on suitable techniques that would provide the basis for analyzing business processes that span a whole supply chain. In particular, we observe a need to present transparent analysis results in a visual and intuitive manner. Process mining is a technique that meets these requirements by generating a graphical process model from a set of event logs of a process [3,4]. Process mining has been successfully applied to various intraorganizational problems [5], but the challenges of using event logs that span different companies is little understood so far.

At the current stage, most RFID scenarios have been implemented to support operational requirements of companies while strategic potentials are mainly neglected (see e.g. [6]). In this paper we contribute to the domain of RFID and process mining research by highlighting in how far the recent EPCglobal standard for exchanging event data can facilitate bridging the gap between data availability and process analysis capabilities. In particular, we focus on a supply chain scenario and illustrate how EPCglobal event types can be used to group event data into cases on a dynamic basis. To be more concise, we sketch an algorithm that assigns events to cases based on the record of EPC events following the EPCglobal standard.

Against this background, the paper is structured as follows. In Section 2 we use an example taken for the SCOR reference model for supply chain operations to illustrate the requirements of handling RFID-event data on an interorganizational level. In Section 3 we present the data available for events according to the EPCglobal standard. Furthermore, we describe an algorithm to assign events to cases based on the event data available in EPCglobal data. In Section 4 we discuss related work. Finally, in Section 5 we conclude the paper and give an outline of future research.

2 Challenges of Mining RFID-Based Supply Chain Processes

In this section we investigate challenges of mining supply chain processes. In Section 2.1 we give a brief overview of RFID capabilities before Section 2.2 highlights the mining challenges by using an example from the Supply Chain Operations Reference Model (SCOR) as an illustration.

2.1 RFID Technology in the Supply Chain

Radio Frequency Identification (RFID) is an automatic identification technology that is used to track location and movements of physical objects. A major advantage of RFID technology in comparison to traditional identification techniques like barcode is that data can be read contactless and as a non-line-of-sight transfer via radio signals [7]. Beyond that, RFID tags enable bulk scanning and rewriting which is not possible with barcodes [8]. The usage of RFID technology requires different components such as tags, readers, middleware, and applications. *RFID tags* are attached to physical objects and store at least a unique identifier of the object that they are attached to. *RFID readers* are the hardware devices that directly interact with the RFID tags. An antenna on the tag emits radio waves generating voltage in the inductor of the passive transponder or triggering the active transponder to send data. The transponder chip starts working with this voltage, uses the inductor as antenna, and sends its ID to the reader antenna in bit-serial form. The major role of the *middleware* is to coordinate a number of RFID readers that are located, for example, at a production line within a single plant or distributed across a whole supply chain. The tasks of the middleware include, among others, buffering, aggregation, filtering of reader data and providing it in a suitable format to *RFID applications* such as include material tracking and tracing, and the identification of work in progress (WIP) materials [7]. Currently, international standards are emerging for the provisioning of RFID data across the internet. The most significant is the EPCglobal standard [9]. It builds on electronic product codes (EPCs) that are made available within an "Internet of Things". This information network, also called the EPCglobal Network, allows producers, suppliers, and consumers to easily access and exchange information about product locations and movements [7]. We get back to details of EPCglobal in Section 3.

2.2 Challenges of Mining Supply Chain Processes

Process mining is a technique for analyzing how processes are executed in reality. The input for process mining is a set of event logs that relate to different instances of a process. These event logs typically stem from process-aware information systems such as workflow applications or ERP systems (see [10,11]). Based on process mining algorithms, it is possible to construct process models from these event logs automatically [3,4]. In contrast to many manually created process models, the models generated in process mining give an accurate account of what is really happening in the company [5]. This would be valuable information also in a supply chain setting. But there are different reasons why the events that are generated by RFID applications cannot directly be used for process mining.

We use the Supply Chain Operations Reference Model (SCOR) [12] to illustrate some of the issues. The SCOR model serves as a reference for designing operations in a supply chain. It is organized in three levels of decomposition. It starts with five primary management processes on the top level (Plan, Source, Make, Deliver, and Return), which are refined on level two according to three different strategies Stock, Order, and Engineer. Level three defines processes for each strategy of level two including inputs and outputs. In particular, we use the SCOR process Deliver-to-order, denoted as D2, as an example (see Figure 1). The inputs and outputs in the figure refer to other SCOR processes (e.g. Source-to-order S2 etc.) [12].

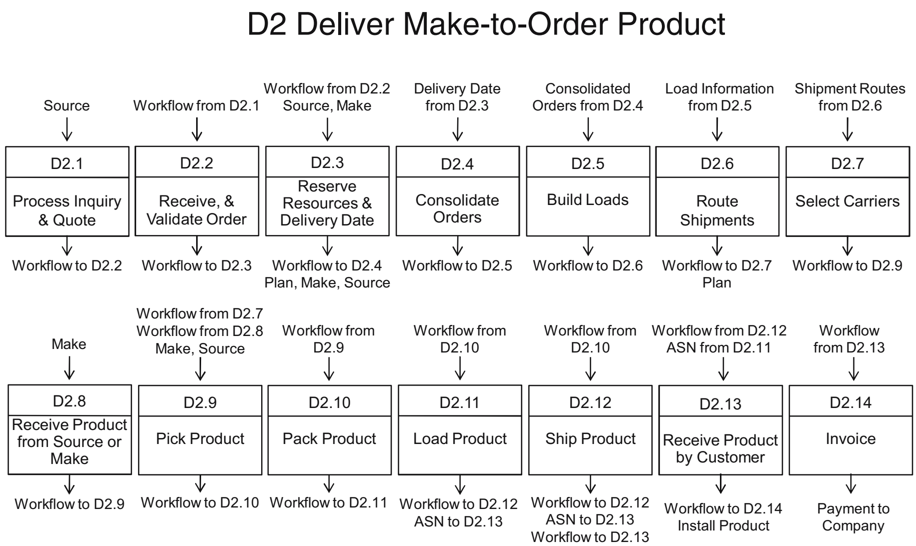


Fig. 1. SCOR Process Category D2

The processing in D2 starts after the receipt and response to general customer inquiries and requests for quotes (D2.1 Process Inquiry & Quote). After the receipt of a customer order the order is entered into an order processing system for validation (D2.2 Receive, and Validate Order) which triggers the reservation of required inventory and planned capacity for specific orders. Furthermore, a delivery date is scheduled (D2.3 Reserve Resources & Determine Delivery Date). If procurable, orders are grouped to ensure least cost or best service fulfillment (D2.4 Consolidate Orders). Afterwards, transportation modes are selected and efficient loads are built (D2.5 Build Loads). In the following step (D2.6 Route Shipments) the loads are consolidated and routed by mode, lane, and location. Afterwards a suitable carrier is selected (D2.7 Select Carriers). After receiving the products, they are verified and put away (D2.8 Receive Product from Source or Make). The pick process contains a series of activities including retrieving orders to pick, verifying inventory availability, building the pick wave, picking the product, recording the pick, and delivering product to packing area in response to an order that belongs to the process (D2.9 Pick Product). Then, after sorting or packing, pasting tags etc., the picked products are delivered to the shipping area for loading (D2.10 Pack Product), where they are placed onto modes of transportation (D2.11 Load Product). The loading process includes the generation of the documentation necessary to meet internal, customer, carrier and government needs (e.g. the advanced shipping notification (ASN)) and triggers the process of shipping the product to the customer's site (D2.12 Ship Product). Finally, the customer receives the shipment and verifies it for completeness and delivery terms. (D2.13 Receive & Verify Product by Customer). The process ends with a signal which is sent to the financial organization that the order has been shipped and that the billing process should begin (D2.14 Invoice).

If we now assume that this whole scenario would be supported with RFID, there are essentially two major problems that hinder a direct evaluation of the process with process mining techniques: case identification and focus shifts.

- The problem with *case identification* stems from the fact that EPC IDs are processed. This means that every event that can be logged only relates to an EPC - there is no explicit notion of a case identifier that groups events belonging to the same process instance. In order to make process mining applicable, we first have to relate every event to a process instance. Using an EPC as a process ID does not work in the general case because of the second problem.
- The problem of *focus shifts* is illustrated in Figure 2. Due to various packing and unpacking, assembly and disassembly operations, it is difficult to follow the object flow of a single process instance. In the example of Figure 2, there is first a packaging step at the shipper's site that includes packing cartons on a pallet. In this process step an aggregation takes place by which the Serialized Global Trade Item Number (SGTIN) is associated with the Serial Shipping Container Code (SSCC) of the pallet on which they are packed. SGTIN serves as an EPC associated with items. The SSCC code is characteristic to containers. Aggregation allows to track which exact products are packed into a container or pallet. Still, the focus on the business object shifts from the single product to the pallet. In the second packing step the pallets are stuffed into a container identified by the Global Returnable Asset Identifier (GRAI). The GRAI contains all SSCC providing the capability to identify and verify individual pallets. The monitoring of each stage of containerization of the product provides the supply chain partners between the manufacturer and the point of final distribution with the ability to understand precisely which products, at a unique pack level, are shipped. This allows e.g. for accurate recall management. Now, the focus has shifted from the single pallet to the container.

There are different ways of dealing with these problems. In the following section, we investigate how cases can be constructed from event logs that follow the EPCglobal standard.

3 Case Construction Based on EPCglobal

In this section, we analyze how the EPCglobal standard can help to make RFID event data available for process mining. In Section 3.1 we introduce the data that is attached to EPCglobal events. Section 3.2 then presents our approach to construct case information from these events.

3.1 Logistic Event Types of EPCglobal

EPCglobal is a nonprofit organization founded in 2003 by the global standardization consortium GS1. Its goal is to define global standards for RFID technology.

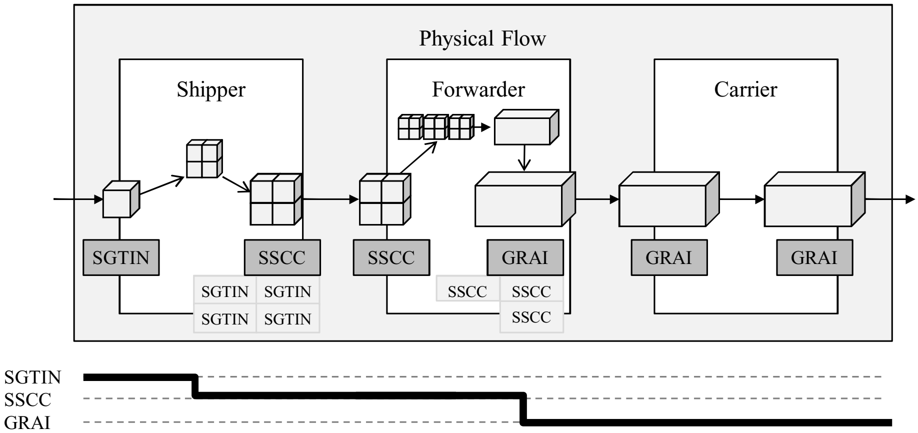


Fig. 2. Packaging Process

The EPCglobal Architecture Framework describes how the different components and standards suggested by EPCglobal will fit together to form the EPCglobal Network. The EPCglobal Network is an information network that will allow producers, suppliers, shipper, and consumers to easily access and exchange information about products. The information access is based on Electronic Product Codes (EPC), these are globally unique identifiers that unambiguously identify each product and can be used as a reference to information stored in the EPCglobal Network. In contrast to barcodes, the EPC identifies each product instance rather than only product categories. One key component is the EPC Information Service (EPCIS). The EPCIS stores the information that can be referenced through the EPCs and is accessible via the Internet [7]. Moreover, the EPCglobal Architecture Framework covers all aspects of reading EPC tag data and participating in the EPCglobal Network including the Object Name Service and the Product Markup Lange. The Object Name Service (ONS) is a global registry of EPC information and the Product Markup Language (PML) is a language specific to EPC data storage and retrieval [8]. For detailed information we relate to [9].

In the context of the EPCs' reading processes the term event is commonly used. EPCglobal defines four *event types* including object, aggregation, quantity, and transaction events. ObjectEvent encompasses information that relates to one or more EPCs. For example, a reader at a warehouse entry door could always record an event when products are moved from the goods entry area to the warehouse. An AggregationEvent is used when objects have been physically aggregated. In this case, numerous objects (e.g. boxes) are put together in a higher-level unit (e.g. pallet). AggregationEvents are created that assign the child EPCs to the parent EPC of the container. AggregationEvents are created recursively if containers contain other containers. For entering inventories a QuantityEvent is used. In this case, the amount is recorded and not the serialized number section. A TransactionEvent describes the assignment of physical objects with one or more transactions. For example, in such an event, the reference to a certain delivery notification could be linked to the objects declared to be inside it [9].

The event types contain *attributes*. These attributes can be divided into four dimensions of information about the object (what = EPC, epcList, transactionList, EPC class, and quantity), the time (when = eventTime), location (where = read point, business location), and the business context (why = process, disposition). The eventTime stores the date when the event is captured. The epcList is a list of EPCs naming the physical objects to which the event pertained. In the case of a transaction event, these objects are linked to the transaction. The parentID represents an identifier of the parent of the EPCs given in the EPC list. The attribute *action* relates the event to the lifecycle of the related EPCs. The attribute can accept the values ADD, OBSERVE, or DELETE. They capture the semantics of the event. Consider the case of a pallet. When it is created by bundling different items, the respective event is an AggregationEvent with action attribute ADD. Once it is disassembled again, e.g. having reached its shipment destination, an AggregationEvent with action attribute DELETE must be reported. The attribute *bizStep* reflects the business step of which the event is part of. The *disposition* shows the business condition of the objects associated with the EPCs. The read point at which the event took place is stored in the attribute *readPoint*. The attribute *bizLocation* references the business location where the objects associated with the containing and contained EPCs may be found. The attribute *type* is an identifier that indicates what kind of business transactions this business transaction denotes. The identifier *bizTransaction* denotes a specific business transaction. The D2 process works as follows if it is supported by RFID technology (see Figure 3). All captured tag data is categorized in terms of the event types ObjectEvent, TransactionEvent, QuantityEvent or AggregationEvent. After the validation of the order the process begins with the production of a product, which is immediately tagged with an EPC tag. The related object event contains information about the EPC, location and time. The subsequent storage of goods is monitored by means of quantity events. The receipt of the single product and the pallet is documented by a RFID reader of the packing site door by means of object events. The following packaging step is linked to an aggregation event and a transaction event, which in turn link the pallet, the cartons on the pallet and the related order. At the packaging site doors and at the shipping site entry door the pallets' move will be captured by means of object events. Prior to passing through the shipping dock door, the pallets are linked to a purchase order and a transport order via a transaction event. Finally, the shipping is monitored by recording an object event. Prior to passing through the shipping dock door, the cartons' movement will be monitored and recorded by means of object events at significant places, such as doorways or shelves. At the shipping dock doors, the EPCs of cartons and pallets are linked to the corresponding advanced shipping notification (ASN) via a transaction event.

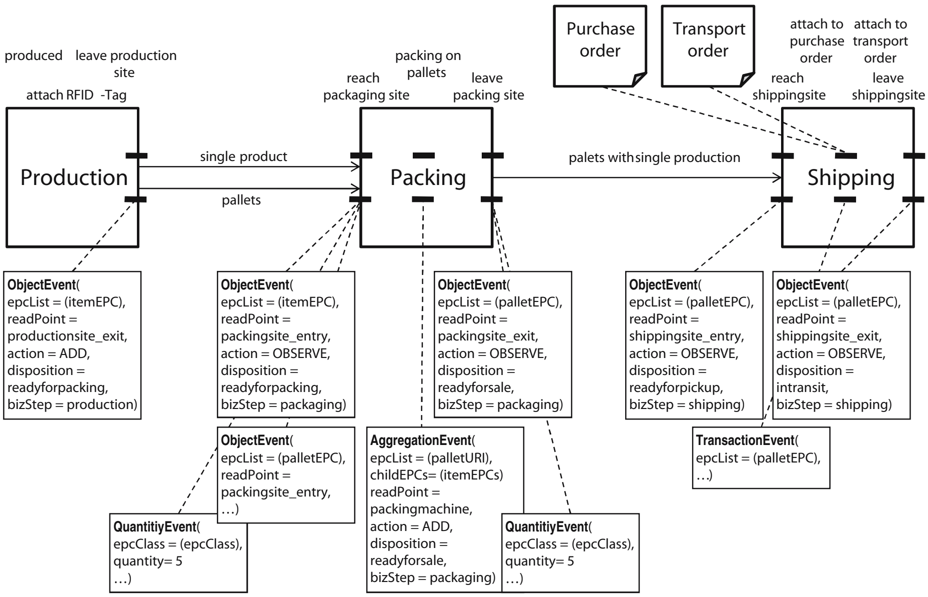


Fig. 3. Internal Process of a Supplier and corresponding Events [13]

3.2 Case Construction Based on EPCglobal Event Types

In order to make RFID event logs accessible to process mining, we have to assign each event at least to a process activity, a time stamp, and a process instance [14]. There are different pieces of information in an EPCglobal event that can be directly mapped to these fields: the *bizStep* attribute translates to the activity that is executed in the process and the time stamp is stored in the *eventTime* attribute. The main challenge is assigning different events and the related EPCs to a process instance in such a way that the shifting focus between different assembled and disassembled business objects of varying granularity is handled appropriately.

Below we sketch an algorithm to construct cases based on assembly (Aggregation add) and disassembly events (Aggregation delete). The key idea of this algorithm is that there are two distinct types of aggregation: production aggregation and shipment aggregation. *Production aggregation* is of a persistent nature: once different pieces are assembled they become a new object of its own that is meant to be persistent. That is different to *shipment aggregation*: for this case the aggregation is of a temporary nature. In a way, it is transparent to the physical production process. Both types of aggregation can be distinguished based on EPCglobal event logs: shipment aggregations occur with both an assembly and a disassembly event while production aggregations only cover an assembly. We use this distinction for our algorithm. Its focus is to construct cases based on physical movement and physical assembly. Therefore, all events that belong to the same production aggregation line are grouped into the same process instance. As a consequence, shipment aggregations are treated to become transparent.

For our algorithm, we can make certain assumptions about the EPCglobal event log. In particular, we assume that the events are in chronological order. This implies that EPCs are in hierarchical order at any point in time due to the fact that the containment relationship (childEPCs) is none overlapping. The goal of the algorithm is to construct a tree of EPCs that belong to the same physical production process. The structure of this tree is built up by links between EPCs. As leaves, there can be different events assigned to an EPC node. We assume objects of type process, of type EPC and of type event to be available. The process object contains the different trees of EPCs which group different events. The objects provide the following methods for constructing trees:

- event.getEPC() to get the EPC of an event,
- event.getChildEPCs() to get the child EPCs,
- EPC.add(event) to add an event,
- process.getEPC(EPC) to retrieve an EPC,
- process.mergeCases(NewRootEPC,List of OldRootEPCs) to group different cases under a new root event,
- process.splitCases(rootEPC) to copy the root EPC to all its child EPCs and delete it.

As input we assume a list of EPCglobal events which we can process in chronological order using an iterator EventChronology. Based on these notations, we can define the algorithm:

1. Set currentEvent to EventChronology.next(),
2. If process.getEPC(currentEvent).exists() then
   (a) If currentEvent.isNotAggregation() then
       process.getEPC(currentEvent).add(currentEvent),
   (b) If currentEvent.isObserveAggregation() then
       process.getEPC(currentEvent).add(currentEvent),
   (c) If currentEvent.isAddAggregation() then
       Set currentEPC to currentEvent.getEPC(),
       process.mergeCases(currentEPC,currentEPC.getChildEPCs())
       currentEPC.add(currentEvent)
   (d) If currentEvent.isDeleteAggregation() then
       Set currentEPC to currentEvent.getEPC(),
       currentEPC.add(currentEvent),
       process.splitCases(currentEPC)
3. Repeat if EventChronology.hasNext(),

For evaluation purposes we have prototypically implemented the algorithm in the context of the ProM process mining framework [15] using the MXML interchange format for log files [14]. This implementation takes EPCglobal event logs as input to generate an MXML file according to the presented algorithm. The challenge for a thorough real-life evaluation is that companies that use the EPCglobal in their logistic processes are still rather scarce. Therefore, we simulated supply chain processes to generate log files using a simulator that builds on top of the EPCIS implementation Fosstrak.[1] By feeding the generated MXML files into ProM and applying a process mining plug-in, we could verify that the approach yields the expected model, also when aggregation and disaggregation operations are included.

4 Related Work

Our work can be related to different streams of research in the business process management domain. Complex event processing [16] is recently receiving increased attention to support process execution in an open and distributed web environment. The challenge there is to assign incoming messages to the correct process instance at runtime. Several executable process modeling languages provide support for this problem by offering so-called correlation sets. A correlation set is essentially a query that retrieves identifiers from messages that are unique for a particular process instance. The correlation set concept is included in BPEL [17] and BPMN [18]. The correlation problem has been discussed theoretically in [19,20]. The idea of a correlation set is somewhat related to our approach. Our algorithm also builds upon the observation that certain identifiers, namely EPCs, belong to a unique process instance. There are some works that relate to mining correlation information including [21] where the authors use an algorithm to identify correlation relevant fields that can be used to construct case identifiers. In [22] the authors define an algorithm to construct a finite state machine of business protocols from logs without case identifiers. In our approach, we reconstruct the correlation relationships that are implicitly given in the EPCglobal event log data.

Our contribution can also be related to practical challenges of process mining. There is a growing body of knowledge that reports on case studies in different application domains. While there are considerable achievements [5], several problems still need to be solved [23]. A particular problem relates to the reformatting and enrichment of log data that stems from information systems that are not directly process-aware. The complexity of this problem becomes apparent in the work on mapping SAP log data to MXML [11]. Our work is unique in this context, as it identifies how RFID event data (available in EPCglobal format) can be used to construct cases such that process mining can be applied.

5 Conclusion

In this paper, we have addressed the current research challenge of making RFID event logs accessible for process mining analysis. We have used an example from the SCOR Supply Chain Operations Reference Model for identifying case identification and focus shifts as major obstacles. Furthermore, we have demonstrated

[1] www.fosstrak.org

how information that is available in EPCglobal compliant RFID event data can be used to construct cases. To be more concise, we have sketched an algorithm to group events to cases. This algorithm handles different types of events including assembly and disassembly events.

We have implemented the algorithm as part of a tool that converts EPCglobal event logs to MXML files. These can be processed with the process mining workbench ProM. From both an industry and a research perspective there are open questions on how the shifting focus can be appropriately supported. Currently, we construct a tree of EPCs belonging to the same process instance in such a way that the most coarse-grained physical object is on top of the tree and all child EPCs are included. In a supply chain, different parties might be interested in following the event stream on different levels of granularity. This challenge requires future research both in terms of mining algorithms but also in terms of interactive tool support. Furthermore, we aim to apply our approach to a real-world scenario. Currently, we are searching for potential partners who already use the EPCglobal standard.

References

1. Lee, H., Padmanabhan, V., Whang, S.: The Bullwhip Effect in Supply Chains. Sloan Management Review 38, 93–102 (1997)
2. Sharma, A., Citurs, A., Konsynski, B.: Strategic and Institutional Perspectives in the Adoption and Early Integration of Radio Frequency Identification (RFID). In: Hawaii International Conference on Systems Sciences, vol. 40, pp. 37–55. IEEE, Los Alamitos (2007)
3. van der Aalst, W.M.P., Weijters, T., Maruster, L.: Workflow mining: Discovering process models from event logs. IEEE Transactions on Knowledge and Data Engineering 16(9), 1128–1142 (2004)
4. Greco, G., Guzzo, A., Manco, G., Saccà, D.: Mining and reasoning on workflows. IEEE Transactions on Knowledge and Data Engineering 17(4), 519–534 (2005)
5. van der Aalst, W., Reijers, H., Weijters, A., van Dongen, B., Alves de Medeiros, A., Song, M., Verbeek, H.: Business process mining: An industrial application. Information Systems 32(5), 713–732 (2007)
6. Ivantysynova, L.: RFID in Manufacturing: Mapping the Shop Floor to IT-Enabled Business Processes. PhD thesis, Wirtschaftswissenschaftliche Fakultät, Humboldt-Universität zu Berlin (2008)
7. Günther, O., Kletti, W., Kubach, U.: RFID in Manufacturing. Springer, Heidelberg (2008)
8. Niederman, F., Mathieu, R.G., Morley, R., Kwon, I.W.: Examining RFID Applications in Supply Chain Management. ACM Transactions on Software Engineering and Methodology 50(7), 92–101 (2007)
9. EPCglobal Inc., EPC Information Services (EPCIS) Version 1.0.1 Specification (2007), `http://www.epcglobalinc.org/standards/epcis/epcis_1_0_1-standard-20070921.pdf`
10. Dumas, M., ter Hofstede, A., van der Aalst, W. (eds.): Process Aware Information Systems: Bridging People and Software Through Process Technology. Wiley Publishing, Chichester (2005)

11. Ingvaldsen, J., Gulla, J.: Preprocessing support for large scale process mining of sap transactions. In: ter Hofstede, A.H.M., Benatallah, B., Paik, H.-Y. (eds.) BPM Workshops 2007. LNCS, vol. 4928, pp. 30–41. Springer, Heidelberg (2008)
12. Supply-Chain Council: Supply Chain Operations Reference Model, SCOR, Version 8.0 (2006)
13. Fleisch, E.: Das Internet der Dinge: Ubiquitous Computing und RFID in der Praxis. Visionen, Technologien, Anwendungen, Handlungsanleitungen. Springer, Heidelberg (2005)
14. van Dongen, B.F., van der Aalst, W.M.P.: A Meta Model for Process Mining Data. In: Pastor, Ó., Falcão e Cunha, J. (eds.) CAiSE 2005. LNCS, vol. 3520, pp. 309–320. Springer, Heidelberg (2005)
15. van Dongen, B.F., de Medeiros, A.K.A., Verbeek, H., Weijters, A., van der Aalst, W.M.P.: The ProM Framework: A New Era in Process Mining Tool Support. In: Ciardo, G., Darondeau, P. (eds.) ICATPN 2005. LNCS, vol. 3536, pp. 444–454. Springer, Heidelberg (2005)
16. Luckham, D.: The Power of Events: An Introduction to Complex Event Processing in Distributed Enterprise Systems. Addison-Wesley, Reading (2001)
17. Alves, A., Arkin, A., Askary, S., Barreto, C., Bloch, B., Curbera, F., Ford, M., Goland, Y., Guizar, A., Kartha, N., Liu, C., Khalaf, R., Koenig, D., Marin, M., Mehta, V., Thatte, S., van der Rijn, D., Yendluri, P., Yiu, A.: Web services business process execution language version 2.0. Committee specification 31 January 2007, OASIS (2007)
18. Object Management Group: Business Process Modeling Notation (BPMN) Specification. Final Adopted Specification, dtc/06-02-01, Object Management Group (February 2006)
19. Barros, A., Decker, G., Dumas, M., Weber, F.: Correlation patterns in service-oriented architectures, pp. 245–259 (2007)
20. Decker, G., Mendling, J.: Instantiation semantics for process models. In: Dumas, M., Reichert, M., Shan, M.-C. (eds.) BPM 2008. LNCS, vol. 5240. Springer, Heidelberg (2008)
21. Pauw, W.D., Hoch, R., Huang, Y.: Discovering conversations in web services using semantic correlation analysis. In: 2007 IEEE International Conference on Web Services (ICWS 2007), Salt Lake City, Utah, USA, pp. 639–646. IEEE Computer Society, Los Alamitos (2007)
22. Serrour, B., Gasparotto, D.P., Kheddouci, H., Benatallah, B.: Message correlation and business protocol discovery in service interaction logs. In: Bellahsene, Z., Léonard, M. (eds.) CAiSE 2008. LNCS, vol. 5074, pp. 405–419. Springer, Heidelberg (2008)
23. Genrich, M., Kokkonen, A., Moormann, J., zur Muehlen, M., Tregear, R., Mendling, J., Weber, B.: Challenges for business process intelligence: Discussions at the BPI workshop 2007. In: ter Hofstede, A.H.M., Benatallah, B., Paik, H.-Y. (eds.) BPM Workshops 2007. LNCS, vol. 4928, pp. 5–10. Springer, Heidelberg (2008)

SOEDA: A Method for Specification and Implementation of Applications on a Service-Oriented Event-Driven Architecture

Matthias Wieland, Daniel Martin, Oliver Kopp, and Frank Leymann

IAAS, University of Stuttgart, Germany
{wieland,martin,kopp,leymann}@iaas.uni-stuttgart.de

Abstract. Both Event-Driven Architecture (EDA) and Service-Oriented Architecture (SOA) are unique architectural styles widely used in today's industries. However, they mostly exist as isolated systems that could greatly benefit from each other when being combined. This paper introduces SOEDA, an approach that unifies EDA and SOA by introducing a development method taking advantage of the unique properties of each architecture. The different steps of the method reach from abstract process specification over event and process implementation to the the final execution phase – described in an abstract manner and by means of an example. Resulting applications are based on state-of-the-art workflow technology using events to trigger the execution of individual business activities.

Keywords: Complex events, EPC, BPEL, transformation, MDA.

1 Introduction

Events are a widely used abstraction to facilitate asynchronous communication in IT systems. Although the terminology varies slightly across different domains, the concept of an event and event communication are omnipresent. Events are used for disseminating information e.g. in mobile environments and sensor networks and for application integration through messaging middleware (cf. Enterprise Application Integration), and monitoring to control and govern large and diverse installations of IT infrastructure.On an abstract level, an event is defined as "any happening of interest in a computer system" [1], e.g. values reported by sensors, timers or generally any detectable state-change that can be described in a computer processable manner. The main characteristic of event-based system is its inherent asynchronous nature, decoupling sender (producer) and receiver (consumer) of an event in the dimensions of time, reference and location. With the rise of Radio-Frequency Identification (RFID) and the ubiquity of computing devices in industry, event-based systems gained a lot of attention in the area of manufacturing and retail e.g. in form of the METRO Future Store[1]. Event-based systems in production today are typically based on Event-Driven Architecture

[1] http://www.future-store.org/

W. Abramowicz (Ed.): BIS 2009, LNBIP 21, pp. 193–204, 2009.

(EDA) [1], an architectural style that is centered around the notion of event and event processing.

Naturally, industries also have a strong interest in Business Process Management (BPM) technology to align and support their business processes with IT infrastructure. Business processes are expressed using specialized languages such as the Web Service Business Process Execution Language [2] (WS-BPEL or BPEL for short) or Event-Driven Process Chains (EPC) [3]. BPEL facilitates process *execution* by providing and standardizing execution semantics for orchestrating business activities. It defines the way in which basic services (business activities in the form Web Services) are used to build new, coarser grained services. E.g. a loan approval process orchestrates the basic services RiskAssessment, CreditCheck and IncomeReview. Since Web Services are an implementation of the SOA architectural style, process systems using BPEL as orchestration language are naturally embedded into an existing service oriented architecture implemented by Web Services. EPCs in contrast are mostly concerned with the modeling aspect of business processes, and therefore put an emphasis on being more easy to use and providing a standardized set of visualization elements, whereas not defining exact execution semantics. This brings them closer to non-technical thinking people who want to concentrate on the high-level process rather than on concrete execution, which stands in direct contrast to what BPEL provides.Typically, transformations are employed on the initial EPC to create a "skeleton" BPEL process that is then further refined by IT staff to make it finally executable.

Clearly missing today is an unified architecture, combining event-driven and service-oriented architectures to one coherent whole. This paper aims at filling this gap by introducing the unified architecture and method SOEDA. The *key advantage* of this method is the integration of the individual strengths of each: EDA-based systems provide the ability to flexibly react to ad-hoc changes, recognize situations, and the power to handle huge amounts of events and data streams; SOA-based systems provide standards compliance, interoperability and legacy system integration. The key element for integrating both approaches is the EPC modeling concept, where events are treated as *"first class citizens"*, i.e. the occurrence of events are fundamental elements of the business process. Each action is always triggered by one or more events; finishing an action again creates events to trigger further actions. This way of triggering actions is transformed to service calls, which are an integral part of the final process execution language: BPEL. The result is a coherent model that uses a *standardized, event-centric business process notation* (in the form of EPC) for *modeling* the initial process, which is then *transformed* into a *service centric execution model* (in the form of BPEL processes) for actual process enactment.

Consequently, this paper is organized as follows: in Section 1.1 we provide an example scenario that is used throughout the paper to exemplify our approach. Related work is discussed in Section 2, which is followed by our system's architecture, given in Section 3. Next, the method, consisting of five steps is discussed in Section 4. Finally, Section 5 summarizes our findings and gives an outlook on future work.

1.1 Motivating Scenario

In this section we introduce an example scenario to motivate in a practical way why the combination of event orientation and service orientation by using the SOEDA method is beneficial for designing workflow based applications. The example is used throughout the paper for a better understanding of the conceptual steps. The scenario given is one possible application field. An example of another field (production order processing) is presented in [4]. In pervasive computing many sensors and actors are deployed in the environment and can be accessed for observing or manipulating the state of the environment. This data can be used in a process as illustrated in the example scenario presented in Fig. 1. Here three processes are shown controlling room temperature to keep it between optimal values and thus forming a closed loop system: If the room temperature is to high the first process starts to lower it. If the room is to cold the second process is started. In the case the temperature exceeds a certain limit for the temperatures it is assumed that the climate system has a defect and the maintenance process is started.

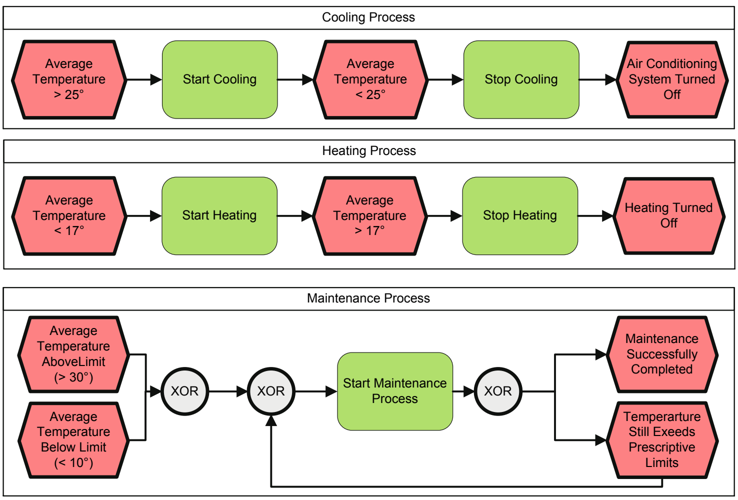


Fig. 1. Example Scenario: Temperature control processes based on sensor observation

Such closed loop systems cannot be managed with a conventional workflow system due to the amount of messages per second (due to the high sampling rates and high number of the installed sensors - each room has many temperature sensors that must be observed individually). Using the concepts described in this paper, BPM is enabled to manage those kinds of processes because stream handling and message reduction is done by complex event processing. Thus, the workflow system receives relevant messages only and can concentrate on handling the control flow.

2 Related Work

Many works are available on combining Service-Oriented Architecture and Event-Driven Architecture (e.g. IBM: [5], HP: [6], Academic: [7]). In summary, they state that it is important to combine these architecture styles. They also state that it is difficult to develop applications for such a combined architecture. As a step towards solving that problem this paper presents in our knowledge SOEDA as the first *method* supporting the *development* of such applications. While there exist many island solutions for parts of our method. SOEDA builds on available solutions where possible.

The Business Process Modeling Notation (BPMN, [8]) is a widely used graphical notation to model business processes. It exists in parallel to the EPC notation. In [9] the Business Event Modeling Notation (BEMN) is introduced; a language with a BPMN-inspired graphical notation to specify events. However, the link to an event infrastructure is not given.

A taxonomy for model transformations is provided in [10]. The most important criteria is the distinction between horizontal and vertical transformation. In a horizontal transformation, the model is transformed to another model on the same abstraction level. In a vertical transformation, the target model resides on a different level of abstraction. In general, transformations of unmarked EPCs are horizontal transformations, whereas transformations of marked EPCs are vertical transformations [11]. A general overview of all available transformations from EPCs to BPEL and their classification using the taxonomy of [10] is given in [11]. In general, all the transformations ignore intermediate events. The only exception is [12], which distinguishes between data-based XOR-splits and event-based XOR-splits, but without stating how events are received from an infrastructure. The most recent work on mapping graphical notations to BPEL is presented in [13]. There, BPMN is mapped to BPEL by building a tree of structures in the graph. The type of each structure can be determined by applying the technique presented in [14]. We reuse these techniques in our approach and adapt it to handle EPCs.

3 High-Level Architecture

Fig. 2 depicts the overall architecture of the proposed SOEDA system: Workflow technology traditionally distinguishes between specification and execution layers, an approach we follow in our architecture. To specify the workflow, designers use event driven process chains (EPCs). Thereafter the domain experts refine the events from the process definition using a domain-specific event description language shown on the right-hand side of Fig. 2 (CEP). Since the process itself is modeled on an abstract level, it must be possible to define events on the same abstract level in order to make it possible for the same person to define the process and the events, without the necessity to know details of either domain.

Since EPCs and the domain specific CEP dialect are not directly executable, algorithms have to be used to transform both representations into executable workflows on the process side and into an executable CEP language on the event side. Event notifications from the event engine are then used to facilitate communication with the workflow execution engine (or with individual workflow instances, to be more precise).

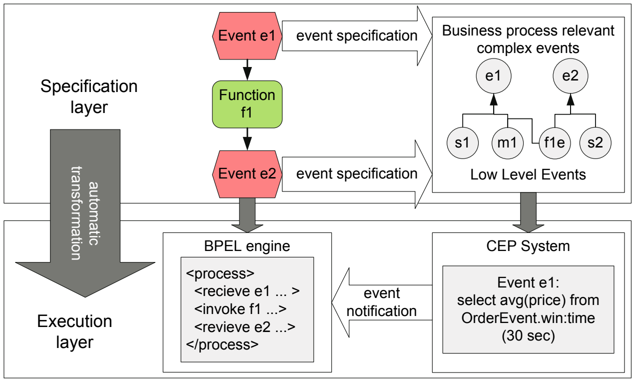


Fig. 2. System Overview

EPCs consist of four main elements: (i) events (depicted as hexagons), (ii) functions (depicted as rounded boxes), (iii) connectors (depicted as circles) and (iv) control-flow arcs. Events in EPCs are passive, i.e. they represent a state change in the system, but do not cause it (they do not provide decisions, but represent decisions taken). Events trigger functions, which are active elements that represent the actual work and again raise events upon completion. Connectors are used to join and split control flow, represented by edges in the EPC graph. An EPC starts and ends with one or more events, process control flow itself strictly follows an alternating sequence of functions and events, possibly with connectors specifying the kind of control flow join and split in between. Further details about EPCs are given in [3].

A BPEL process defines an orchestration of Web Services and consists of structured and basic activities. The actual business functions are not realized in BPEL itself, but by Web Services, where the business data is sent to and received from services using messages. Hence, the most important basic activities are `invoke` and `receive`. An `invoke` activity is used to send a message to a Web Service. A `receive` activity is used to receive a message. The structured activity `pick` realizes an one-out-of-m choice of messages to receive: the first arrived message wins and the other messages are ignored at that activity. Control flow itself is either modeled block-structured using `if` and `sequence` activities or using graph-based constructs realized by the `flow` activity. The graph gets an intuitive execution semantics by marking non-reachable activities and paths as "dead". The execution semantics itself is called "Dead-path Elimination", which is formally defined in [15], specified for BPEL in [2] and explained in detail in [16]. A BPEL process does not need to be executable by itself. The BPEL specification offers to model *abstract processes*, which may hide operational details. One type of abstract processes are *templates*, where so called `opaque` activities can be used to model left-out behavior. We see these templates as abstract workflows. For a formal definition of EPC and BPEL syntax we have to refer the reader to the technical report [4] due to of space limitations in this paper. The necessary definitions regarding events are presented in Definition 1.

Definition 1 (Event Sets)

- *Set of complex event names* $Œ_N$ *consists of all names of complex events available in the system. They are identical to the names of the EPC events.*
- *Set of complex event definitions* $Œ_D$ *consists of descriptions for the deduction of all complex events using event adapters and conditions.*
- *Set of complex event queries* $Œ_Q(CEPsystem)$ *contains concrete implementations of all* $œ_d \in Œ_D$ *for a specific complex event processing system, e.g. Esper*[2].
- *Set of event sources* E_S *contains all sensors or other raw source events producers.*
- *Set of event adapters* E_A *contains a set of Web Services that produce output only and make event sources technically accessible.* E_A *publishes events of an* E_S *which are received in a BPEL flow.*

4 SOEDA Method

This chapter presents the main contribution of the paper: A development method for workflow based applications based on event-driven service-oriented architectures. Due to different semantical layers and architecture styles, the development of such applications is very complex. To handle the complexity the SOEDA method builds on the MDA (Model-Driven Architecture) software design approach. That means the models are transformed from an computational independent model (CIM) to an platform independent model (PIM) and after that to an platform specific model (PSM). By that SOEDA can help saving development time, because the method provides as much as possible automated transformation for the different steps in Fig. 2. The main benefit results from the automatic transformation of the EPCs semantic business process specification to the very detailed abstract BPEL workflow artifacts.

Following subsections describe each transformation step of the method in detail. Here the CIM is represented by the EPC model because it is mainly natural language description for human to human communication (step 1). The PIM model is represented by the abstract BPEL code (step 3) and the wiring descriptions for the events (step 2, 4). The PSM model at the end results from the transformation of the PIM using a concrete platform model (PM). In our case the PSM consists of the executable BPEL workflows, the deployment descriptors for a BPEL engine and the executable CEP queries e.g. for Esper (step 5). All examples shown in the following use the example EPC "Maintenance Process" in Fig. 1.

Step 1: Process Definition: The first step is the creation of the EPC specification by domain experts. For this task any EPC modeling tool, such as the ARIS toolset or Oryx[3], can be used. We propose to use EPC as high level process definition language because it provides events as an integral part. That has the

[2] `http://esper.codehaus.org/`

[3] `http://www.oryx-editor.org/try`

big advantage in contrast to other modeling languages like for example BPMN that an alternation between event based processing and workflow based service oriented control flow is prescribed by the process model. *Example for Step 1:* A sample EPC model is presented in Fig. 1.

Step 2: Complex Event Extraction: The second step is the automatic extraction of all event names from the EPC specification (E_{EPC}). Event names are used as names ($C\!E_N$) for the complex event definitions ($C\!E_D$) that are defined in step 4. The complex events are deducted from the available event sources. In the execution phase these events are the messages the workflow is waiting to be notified of in a receive activity.

Example for Step 2: $C\!E_N = E_{EPC} \iff C\!E_N =$ {AverageTemperatureBelowLimit, AverageTemperatureAboveLimit, TemperartureStillExeedsPrescriptiveLimits, MaintenanceSuccessfullyCompleted}

Step 3: Process to Workflow Transformation: The third step is the transformation of EPC control flow to an abstract BPEL flow. A main goal of the transformation is that both control flows have a similar structure to enable domain experts to understand the executed BPEL workflows due to the similarity to the EPCs they modeled as specification. This allows business activity monitoring to be done directly in the BPEL flow without additional efforts. *The main transformation idea is to map EPC functions to opaque activities and EPC events to message receive activities.* Each `opaqueActivity` has to be manually refined in the executable completion. The message receiving constructs are directly executable.

Graphs are enforced to be acyclic in BPEL workflows. To express loops, the `repeatUntil` and `while` activities are offered. In addition, a choice between multiple incoming messages has to be modeled by a `pick` activity. In contrast to a BPEL workflow, an EPCs is allowed to be a cyclic graph. To map EPCs to BPEL workflows, we keep the original graph structure as much as possible in the resulting BPEL workflow. To detect the structures which have to be mapped to the block-structures offered by BPEL, we combine the techniques presented in [13] and [14]. [13] presents a technique to identify the structure of a graph, called "Process Structure Tree" (PST). The structures of the tree can be classified as types such as repeat-until-loop by using the technique presented in [14]. In case a structure cannot directly be expressed using BPEL constructs, we map this structure into an `opaqueActivity`. We use transformation patterns to transform identified structures. A transformation pattern consists of the source EPC structure and target BPEL structure. In the following, we present the mapping concept as well as one example transformation pattern. Further patterns and technical details are presented in [4].

The general idea is presented in Fig. 3: the EPC function f1 is transformed to an `opaqueActivity` and the event e1 is transformed to a `receive`. The `opaqueActivity` and the event are connected using a link $l1$, which takes no transition condition. The name of the link is made unique within the BPEL process. The name of the

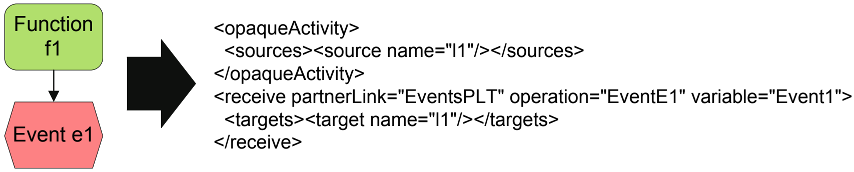


Fig. 3. General idea of the transformation

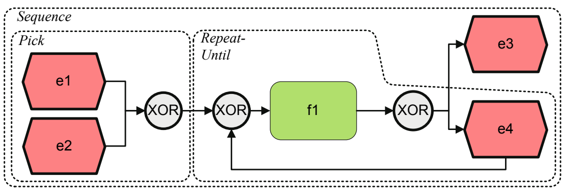


Fig. 4. Process Structure Tree for the Maintenance Process

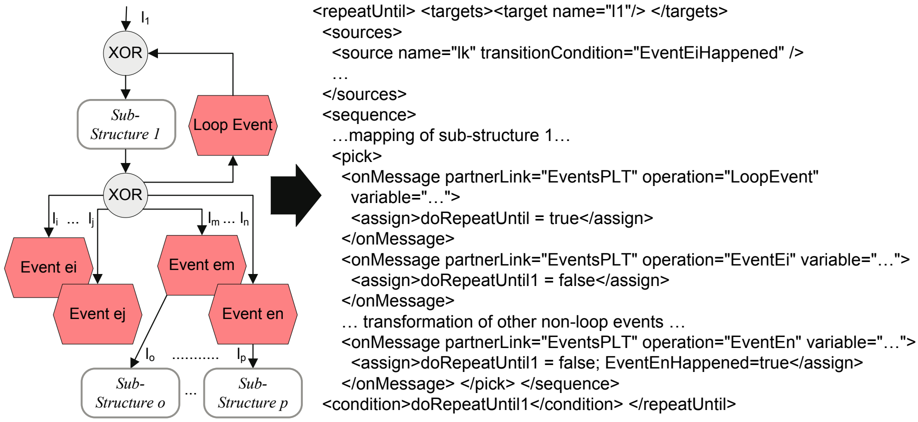


Fig. 5. Repeat-until transformation pattern (the syntax of the assign activity is simplified)

BPEL `opaqueActivity` is the camel case version of the name of the EPC function. The `receive` does not get a name assigned. The name of the partner link is always `EventsPLT`. The name of the operation is the camel case version of the event. The name of the output variable is `Event<i>`, where `<i>` is replaced by an unique number.

If an event is used by a transformation pattern, the mapping of that event is defined by that pattern, as in the case of the start events and the repeat-until loop used in the maintenance process (Fig. 1). To determine structures, a process structure tree [13] is built (Fig. 4). The identified sequence is transformed to a `flow` activity, where the contained activities are connected using `links`. The pick-block at the beginning is transformed to a `pick` activity. Repeat-until loops are transformed as shown in Fig. 5: to indicate whether the loop has to be run, the indicator variable `doRepeatUntilLoop1` is used. Three types of events are

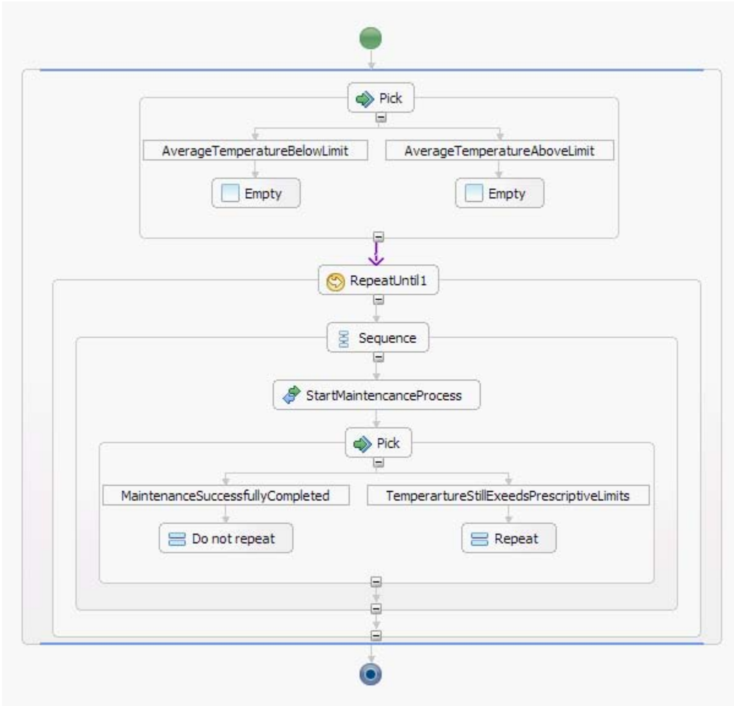


Fig. 6. Eclipse BPEL editor view of "Template-MaintenanceProcess.bpel"

distinguished: the loop event (Loop Event), end events (Events ei to ej) and intermediate events (Events em to en). In contrast to end events, intermediate events have successors. Each event is catched by a `pick` activity. If the event is the loop event, the indicator variable is set to `true`. If the event is an end event, the indicator variable is set to `false` and no further transformation action has to be taken. If the event is an intermediate event, the indicator variable is set to `false` and a Boolean variable (`EventEnHappened`) is set to `true`. This variable is used as transition condition on the `link` connecting the `repeatUntil` with the mapped sub-structure belonging to the event.

Example for step 3: The generated BPEL template for the "Maintenance Process" is shown in Fig. 6, which is a screenshot of the BPEL file opened in the Eclipse BPEL designer[4]. The two start events of the EPC are transformed to an BPEL `pick`, because they are connected by an "exclusive or". The loop is represented in BPEL as `repeatUntil`, since it starts with an activity. It is important to note that the transformation is a vertical transformation, in the sense that not only the control flow structure is mapped, but also the events are enriched by a connection to Web Services.

[4] `http://www.eclipse.org/bpel/`

Step 4: CEP Rules Specification: Wiring of source-events to complex events, i.e. the specification of the rules how complex events are created, is defined using a dedicated event wiring tool. The resulting specification is targeted to communicate these complex event generation rules to a developer that implements code for the event system that aggregates basic events to the complex events with the semantics that were specified. Generally, the event definition is always a tree with the complex event as the root node and the source events or other complex events as leave nodes. Intermediary nodes are operators. Up to now a language such as BEMN [9] could be used. However, the event wiring tool also has to specify how correlation is initiated in the case of a start event and how events and process instances are correlated using e.g. a correlation ID. Please note that it is not necessary to have a dedicated modeling tool at hand in order to be able to carry out this task. Any graphical modeling tool that is able to draw simple graphs and annotate them with textual description suffices. Natural language is used for the definition of all $Œ_D$: Therefore, 1. Each E_S needs to be identified, i.e. it must be clear which actual source of events (a sensor for instance) should be used. 2. Each arc has to denote a condition that is used to filter events flowing through. 3. OP may be any kind of functionality, e.g. a logical operator or set operation that performs actions on all incoming events. 4. Each complex event $Œ_N$ therefore is specified through basic events and a combination of operators. All information necessary to deduct the complex event must be included.

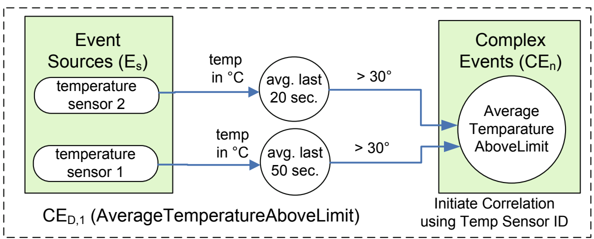


Fig. 7. Wiring of Source-Events to Complex-Events

Example for step 4: Fig. 7 shows an example wiring. Here event streams from two different temperature sensors (1 and 2) are emitted , filtered (for C°), aggregated (for 20 and 50 sec.), and finally filtered again (average temp >30°). Every event that passes through produces an "AverageTemperatureAboveLimit" complex event.

Step 5: Executable Completion: The last step is a purely technical step. This step should be done by IT experts hence only technical execution information has to be inserted. On the one hand all $Œ_D$ definitions must be mapped to queries for the used CEP system ($Œ_Q$). Also event adapters E_A, possibly employing Common Base Events (CBE, [17]), for all needed event sources E_S have to be provided (installed or implemented). On the other hand, the generated BPEL flow is a template which means it cannot be executed directly. The missing execution information and refinements needed for execution have to be added by the IT-expert: substitute all opaque activities, define variables and assigns for the process internal data flow, and last but not least selecting the Web Service interfaces (WSDL) that should be used in the `invoke` activities.

Example for step 5: An complete example with executable BPEL code and WSDL definitions can be found in [4]. Due to space limitations it is not possible to go into further details in this paper.

5 Conclusion and Future Work

We presented SOEDA, a new method that helps to cope with the yet unsolved challenge how to specify complex events and their effects on workflows. The method was illustrated by an example. Further prove for applicableness of the proposed method is shown in the technical report [4] by presenting a complete walkthrough of all steps based on two examples of contrary application areas. Regarding the mapping of EPC to BPEL, our work combines the work of [13] and [14] to provide a complete mapping with support of loops and workflow start events.

Future work is to do research on how provide good tool support for the presented method. There are different areas of tools that would help a domain expert to use the proposed method: first an integrated modeling tool that allows to model EPC processes, BPEL processes and transformation patterns. Secondly a complex event wiring tool that allows the specification of the complex events based on the available sensors.

References

1. Muehl, G., Fiege, L., Pietzuch, P.R.: Distributed Event-Based Systems. Springer, Heidelberg (2006)
2. OASIS. Web Services Business Process Execution Language Version 2.0., (April 2007) (retrieved at 14.8.2007), `http://docs.oasis-open.org/wsbpel/2.0/wsbpel-v2.0.html`
3. Scheer, A.-W., Thomas, O., Adam, O.: Process Modeling Using Event-Driven Process Chains. In: Process-Aware Information Systems: Bridging People and Software Through Process Technology, pp. 119–146. Wiley & Sons, Chichester (2005)
4. Wieland, M., et al.: Events Make Workflows Really Useful. Technical report, University of Stuttgart, IAAS, Germany (2008), `http://www.informatik.uni-stuttgart.de/cgi-bin/NCSTRL/NCSTRL_view.pl?id=TR-2008-09&engl=1`
5. Maréchaux, J.-L.: Combining Service-Oriented Architecture and Event-Driven Architecture using an Enterprise Service Bus. In: IBM Developer Works (2006)
6. Wei, M., Ari, I., Li, J., Dekhil, M.: ReCEPtor: Sensing Complex Events in Data Streams for Service-Oriented Architectures. Technical report, HP (2007)
7. Laliwala, Z., Chaudhary, S.: Event-driven Service-Oriented Architecture. In: ICSSSM. IEEE Computer Society, Los Alamitos (2008)
8. Object Management Group. Business Process Modeling Notation, V1.1 (2008)

9. Decker, G., Grosskopf, A., Barros, A.: A Graphical Notation for Modeling Complex Events in Business Processes. In: EDOC. IEEE Computer Society, Los Alamitos (2007)
10. Mens, T., Van Gorp, P.: A Taxonomy of Model Transformation. Electronic Notes in Theoretical Computer Science 152, 125–142 (2006)
11. Stein, S., Kühne, S., Ivanov, K.: Business to IT Transformations Revisited. In: MDE4BPM (2008)
12. Ziemann, J., Mendling, J.: EPC-Based Modelling of BPEL Processes: a Pragmatic Transformation Approach. In: MITIP (2005)
13. Vanhatalo, J., Völzer, H., Koehler, J.: The Refined Process Structure Tree. In: Dumas, M., Reichert, M., Shan, M.-C. (eds.) BPM 2008. LNCS, vol. 5240. Springer, Heidelberg (2008)
14. García-Bañuelos, L.: Pattern Identification and Classification in the Translation from BPMN to BPEL. In: Meersman, R., Tari, Z. (eds.) OTM 2008, Part I. LNCS, vol. 5331, pp. 436–444. Springer, Heidelberg (2008)
15. Leymann, F., Roller, D.: Production Workflow: Concepts and Techniques. Prentice Hall PTR, Englewood Cliffs (2000)
16. Curbera, F., et al.: Exception Handling in the BPEL4WS Language. In: van der Aalst, W.M.P., ter Hofstede, A.H.M., Weske, M. (eds.) BPM 2003. LNCS, vol. 2678, pp. 276–290. Springer, Heidelberg (2003)
17. Bridgewater, D.: Standardize messages with the Common Base Event model. In: IBM DeveloperWorks (2004)

Enhancing Semantic Service Discovery in Heterogeneous Environments

Jannis Rake, Oliver Holschke, and Olga Levina

Technische Universität Berlin, Fachgebiet Systemanalyse und EDV, Franklinstr. 28-29, 10587 Berlin, Germany
{Jannis.Rake,Oliver.Holschke,Olga.Levina}@sysedv.tu-berlin.de

Abstract. The advent of service-oriented architectures promises companies efficient and effective support of their business processes. Discovery of services with appropriate functionality may be supported by enhancing the syntactical information using additional semantics. However, services of different providers are generally described with heterogeneous concepts. This paper investigates how heterogeneity during service discovery may be overcome using automatic similarity computation. A suitable approach is introduced and a prototype demonstrates that a reliable alignment tool increases overall quality of service discovery.

Keywords: Service Discovery, Semantic Web Services, Ontology Alignment, BPM Integration.

1 Introduction

Service-oriented architectures (SOA [1]) are increasingly gaining momentum. Many companies hope to increase the flexibility of their IT-systems while at the same time reduce costs and time for new development projects by harnessing the promised reusability of services. The foundation of these promises is partly based on the assumption of having a multitude of service providers hosting a wide range of services with necessary functionality. These conditions would generally allow new functionality to be created by simply choosing and combining the right services while keeping the need for customization to a minimum. However, a fundamental prerequisite for reaping these benefits is an efficient and effective way of discovering the services with the appropriate functionality. To aid this service discovery process in the future, semantic service descriptions promise to increase the quality of discovery results and deliver service compositions (semi-)automatically [2]. Existing approaches in the area of semantic service discovery generally assume homogeneity of knowledge bases used for the service descriptions. In a real world scenario such an assumption seems rather unrealistic. Other initiatives that are dealing with heterogeneity of the descriptions unfortunately do not detail how mappings are established or how much (manual) effort is connected with the construction of these mappings.

This paper aims at answering the question, what implications on result quality are connected with the use of automatic mappings between heterogeneous service concepts. As a hypothesis we assume that automation may lead to false positives on the

W. Abramowicz (Ed.): BIS 2009, LNBIP 21, pp. 205–216, 2009.

one hand, while on the other hand increases the amount of true positives, thus increasing overall result quality while minimizing the manual effort of constructing mappings. To ensure future applications in the real world and justify applications in industry, services descriptions of real services are being used. The services are provided by two companies that collaborate in a joint research project on service discovery. The work at hand is based on the approaches that were developed in this project. Additionally, the concepts for service descriptions have been extracted based on knowledge bases of the two providers.

The paper is structured as follows: chapter 2 introduces the fundamentals of semantic services, their descriptions and semantic service discovery. Subsequently, approaches for similarity computation and representation in the area of ontologies (used as knowledge bases) are described and their potentials for overcoming heterogeneities in service discovery are analyzed. In chapter 4 the authors present a solution for the discovery of semantic services in a heterogeneous environment and the implementation of the solution. Based on the implementation an evaluation of the solution is conducted and documented in chapter 5. Following an investigation of the state-of-the-art in this research area, the paper concludes with a discussion of the results and an outlook on future research issues.

2 Semantic Web Services (SWS) and Discovery

The often cited Semantic Web of Berners-Lee is based on the idea of making information of the World Wide Web machine-readable and thus machine-processable. To make this vision become reality existing data needs to be annotated with additional information to convey its meaning, so called semantic annotations [3]. Especially in the domain of Web Services it is desirable to describe the functionality of interfaces with semantics. A description solely based on syntax via WSDL files does not suffice to (semi-)automatically discover, compare and combine services which would create real added value [2]. Numerous semantic description approaches have been developed. The most prevalent are being discussed in the following.

2.1 Semantic Service Descriptions

The Web Ontology Language for Services (OWL-S) is based on the W3C recommendation OWL (Web Ontology Language, [4]). OWL is being used to create an Upper Ontology for services thus providing a meta-model for service descriptions. OWL-S consists of three sub-ontologies: profile, process model and grounding. The most comprehensible way to describe the functionality of a service is realized by the profile, based on a formalization of inputs and outputs as well as precondition and results (IOPR). [5]

The Web Service Modeling Ontology is essentially a meta-model for aspects related to semantic Web Services, including definitions for the structure and semantics of meta-data. Based on four concepts – ontologies, web services, goals, and mediators – WSMO's goal is to enable the automation of Web Service integration, including discovery, composition, mediation, and execution, among other tasks [6]. The capabilities of abstract Web Service descriptions are characterized primarily by pre- and post-conditions as well as assumptions and effects. All of these elements are designed as axioms in WSML (Web Service Modeling Language), allowing special reasoning capabilities. These descriptions allow, in addition to simple keyword searches (which reside more on a syntactic level), simple or complex semantic discovery [7].

SAWDL is a continuation of the W3C member submission WSDL-S (WSDL-Semantics) [8] that aims at providing a lightweight approach to add semantics to Web Services. It is not a formal description language to express semantic models but it rather extends the existing WSDL notation with references to such models. This is done by introducing extensions describing inputs and outputs as well as preconditions and effects. The models referenced by these extensions are maintained outside the WSDL-Files. Such an ontology agnostic approach allows the use of different ontology languages [9, 10].

2.2 Semantic Service Discovery

Service discovery plays an important role in service-oriented architectures. To enable reusability of existing services formal descriptions have to exist, which, during the process of service discovery, can be compared to the requirements of a service consumer. Semantic service discovery is more concerned with the description of functionality and less with how to communicate with the service interface. [11]

Generally, service discovery does not necessarily equal Web Service discovery, as depicted in the abstract process of service discovery in Fig. 1.

Fig. 1. Service Discovery Process cf. [7]

A discovery process starts with the desire to find a service that satisfies certain conditions. Desire is translated into goals (depending on the capabilities of goal formulation). To do so, one has to abstract the goals from the desires to make a service discovery possible. At this point, depending on the specific approach, differences in the possibilities and limitations for goal definition may exist. In a particular environment, only certain characteristics of goals can be expressed and thus searched for. Web services can now be discovered to fulfill or partially fulfill the goal. Depending on the capabilities of the discovered Web Services, refinements need to be made to create a service (e.g. composition) that fulfills the desire at the beginning of the process.

Abstraction of the actual (Web) service plays an important role. Existing approaches for semantic service descriptions generally utilize formal semantic models that essentially limit the vocabulary of the description. Dealing with heterogeneity of these semantic models (e.g. in form of ontologies) is a major challenge during service discovery: even though services share the same functionality, their descriptions may use concepts from different models.

To deal with such heterogeneity, only one of the abovementioned approaches explicitly provides a concept. WSMO defines different kinds of mediators, including ontology to ontology mediators, aiming at providing terminological interoperability between the concept of these ontologies [12]. Although SAWSDL allows referencing schema mappings, this approach provides certain interoperability of data only, but not interoperability of different ontologies [10]. OWL-S provides no explicit concepts for mediation [13].

Nevertheless, mediation between concepts of heterogeneous ontologies is only possible, if similarities between these concepts can be identified and mappings specified. Dealing with this challenge is the central theme of the following section.

3 Ontology Alignment and Mapping

The identification of potential similarities between two ontologies can be represented by an alignment. To provide a common undestandiong for the remainder of the paper, we define ontology alignment according to [14] as follows:

"Given two ontologies, aligning one ontology with another one means that for each entity (concept, relation, or instance) in the first ontology, we try to find a corresponding entity, which has the same intended meaning, in the second ontology. An alignment therefore is a one-to-one equality relation. Obviously, for some entities no corresponding entity might exist."

The process necessary to find the relations or corresponding entities, respectively, is generally called matching. An alignment is manifested in a mapping which generally includes rules for transforming entities from one ontology to entities of another ontology. [15]

Discovery of similarities between the two ontologies via matching may be conducted manually, semi-automatically or automatically. Numerous tools exist for computing such similarities automatically, all based on different strategies and algorithms. Similarity-computation is driven by structural as well as linguistic or semantic measures. Possible techniques in this area include the construction of virtual documents [16], formalizing mappings as decision problems to make use of Bayesian networks [17] or the analysis of semantic sub-graphs [18].

The real benefit of these tools can be seen once their results, the actual alignment definitions, can be reused in other areas of research, such as in knowledge management systems or during service discovery. However, the decision which tool should be used for a certain use case makes it necessary to have a common base of comparison. Established in 2004, the Ontology Alignment Evaluation Initiative (OAEI) provides such a base. Its goal is to compare existing systems and algorithms for ontology alignment. By analyzing the results of these comparisons the best strategies for matching can be identified. Such comparison is made possible by the definition of numerous benchmark tests, which, classified into categories, address specific aspects of heterogeneity between ontologies. The annual contest has seen 18 participants in 2007 (up from 4 in 2004) and a steady improvement in test results and performance of the systems being observed. [19]

The possibility of comparing these systems is made possible, apart from a standardized alignment format, by the implementation of an Alignment API. This API allows standard functions such as loading and serializing alignments as well as more sophisticated methods like the definition of thresholds and automatic comparison of alignments. [20]

The standardization of the ontology alignment definition format and the functionality of the Alignment API open up possibilities for the abovementioned reuse in other areas. For example, it may be beneficial to create an interface for different matching tools to be used in the domain of service discovery. Such a solution would provide the means to evaluate the benefits and drawbacks of different tools for this special use case. This way, those systems that fit a use case best can be identified, and, if necessary, also be replaced by newer versions that provide even better results. The following section investigates the implementation of such an approach.

4 Framework and Architecture

This section describes a framework and an implemented system for the discovery of services in an environment of heterogeneous ontologies. The architecture consists of five parts: discovery component, reasoner, service & ontology repository, ontology alignment component and integration into a business process modeling (BPM) environment (see Fig. 2).

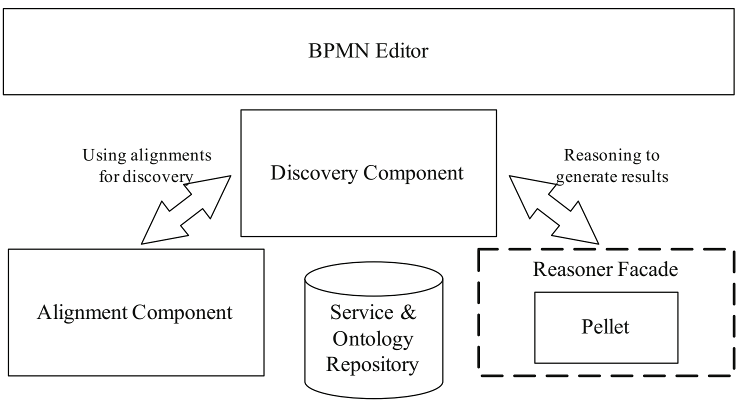


Fig. 2. Architecture of the implementation

4.1 Service and Ontology Repository

The repository holds all service descriptions, on which a search request can be performed during the discovery process. The descriptions are formalized by WSMO Web Services. Additionally, an important enhancement was made to the descriptions. Many other approaches only use one or more pre- and/or post-conditions for each service. An assumption that is closer to reality is thus neglected, the fact that services generally consist of more than one operation, that may need different descriptions via pre- and post-conditions. Our approach allows a discovery per operation using a formalized identification of pre- and post-conditions via suffixes that can be interpreted by the service discovery component.

The pre- and post-conditions are described using logical expressions in WSML-DL (description logic). Currently, services of one provider are solely described by concepts that exist in that provider's ontology. The service descriptions in the repository represent real services of the abovementioned providers, thus demonstrating the feasibility in a real world scenario. Additionally, the repository may hold all necessary ontologies. As an alternative, ontologies may also be loaded dynamically when needed, provided that a valid URI exists and is accessible.

4.2 Reasoner

A connection to a reasoner is made possible by using the WSML2Reasoner component, which, among other functionality, provides a façade for different already established reasoners. This allows the usage of popular reasoners like Racer, Pellet or Fact++. The current implementation uses Pellet [21]. Future evaluations concerning the performance of the implementation are aided by the fact, that a simple exchange of reasoners is made possible by the façade.

4.3 Ontology Alignment Component

A major system requirement is the ability to find all matching services following a service request, while not depending solely on the ontology used by that request. Preconditions for such a discovery are: a) concepts with the same intended meaning exist in the ontologies, and b) these concepts are used to create service descriptions and search requests.

The alignment component makes use of the abovementioned Ontology Alignment API. This not only allows the computation of ontology alignments using the specified ontology matching systems. It is also possible to convert these alignments into valid OWL-axioms. Currently, it is not possible to render the alignments directly into WSML expressions. Therefore, the component creates OWL axioms which can be either registered directly at the Pellet reasoner or may be translated into other languages as necessary. The axioms describe equality relations between the concepts of the ontologies that were identified by high similarity measures, thus allowing enhanced reasoning capabilities. Alignments are computed automatically and used without any manual controls. Benefits and drawbacks associated with this design are being discussed in chapter 5 and 6. How to make use of the alignment is detailed in the next section.

4.4 BPMN-Editor Integration

The service discovery is integrated into a freely available BPMN (Business Process Modeling Notation [22]) editor. This integration allows users to be supported in their formulation of search requests. Information of semantically annotated business processes can be extracted and used as recommendation for the creation of pre- and post-conditions necessary for the request. Additionally the user is being aided by easy navigation in the ontologies using auto completion features similar to [23]. Once the necessary concepts have been chosen by the user, a WSMO goal is constructed automatically, which is then used to start the discovery process. Once the process is completed, the results are presented to the user inside the editor environment. Based on the result list the user may refine his search request or choose service operations to support certain activities within the business process (operations may be dragged from the result list and dropped onto the activity they should support). This allows future conversions into executable processes. The integration thus supports the steps of *goal discovery* and *refinement* from Fig. 1.

4.5 Discovery Component

The discovery component is the core of the architecture. It allows matchmaking of service requests with service descriptions in the service repository. Upon initialization all necessary ontologies are registered at the reasoner component. In case the use of automatic alignment is activated, a request for creating such an alignment is send to the alignment component. The resulting axioms are subsequently also registered with the reasoner, thus updating its knowledge base.

The actual discovery process begins once service requests are created via the extended BPMN editor. Requests are formalized as WSMO goals using pre- and/or post-conditions in WSML-DL. Afterwards, a search process is activated which matches the descriptions of service operations with the requests using the reasoner component. The underlying algorithm is not discussed in detail in this paper, due to limited space. The search process results in three kinds of hits as already defined by other approaches for service discovery [10]:

Exact: Service request and service description are equal or a subsumption exists in both directions, respectively.

Plug-in: The service request is subsumed by the service description, i.e. the service provides at least the requested functionality.

Subsume: The service request subsumes the service descriptions, i.e. only a subset of the requested functionality may be provided by the service.

Additional results can be classified as intersects or disjoints. These results are considered irrelevant for the presentation to the end-user, since they may not be used to fulfill the requested functionality, not even partly. The list of relevant results allows the user to refine the requests or choose services for further usage.

Since architecture and functionality of service discovery have now been clarified, the following chapter aims at critically evaluating which benefits and drawbacks are associated with using automatic ontology alignment during the service discovery process.

5 Evaluation

To evaluate the usage of automatic alignments two exemplary experiments were conducted:

Experiment 1: Service Discovery without alignment
Experiment 2: Service Discovery with automatic alignment

The quality of discovery is being measured using established metrics of information retrieval, namely:

Recall: Number of relevant operations[1] in the result list over the overall number of relevant operations in the collection,

Precision: Number of relevant operations in the result list over the number of operations in the result list, and

F-Measure: Combination of Recall and Precision using a weighted harmonic mean. [15]

For experiment 2 an automatic alignment is being created using the matching systems FalconOA [16] and LILY [18], both of which have shown very good results in the benchmarking track of the OAEI 2007 [19]. Evaluation was conducted based on four requests and on a total of 22 services including 41 operations of two service providers. The results for the experiments are depicted in Fig. 3. To simulate the implications of increased heterogeneity the ontology of the second service provider was altered in the style of OEAI benchmarks. One alteration included the introduction of synonyms while the other alteration consisted of removing all vowels from the concepts to test on structural similarity detection.

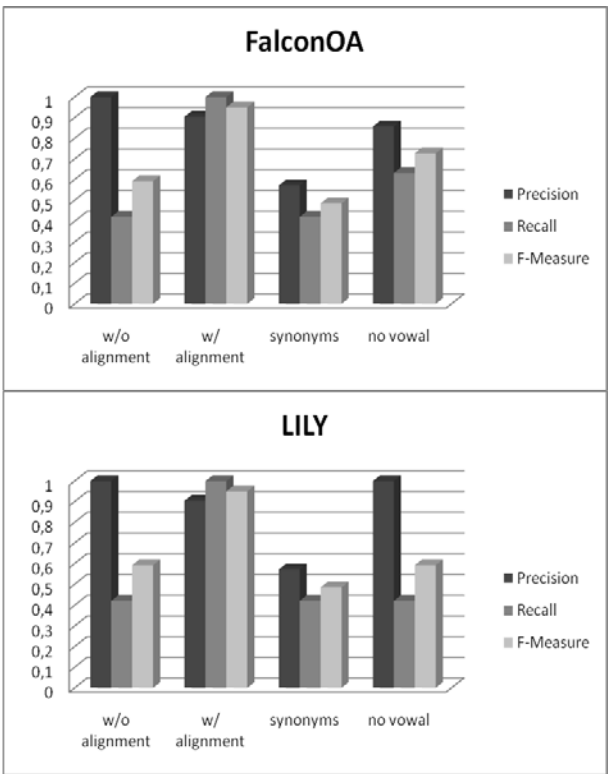


Fig. 3. Evaluation of service discovery with and without alignment

[1] The evaluation is conducted based on operations since the discovery allows results to be displayed on a per-operation-basis.

The evaluations show that without any alignments all relevant services of the first provider were found. This was to be expected given that these were used to create the requests. This first test yields a very high precision but a poor recall, since the services of the second provider could not be found. Once we introduce the automatic alignments one can see a drastic increase of recall at the cost of a minor decrease in precision, which effectively leads to an overall increase in F-measure. These results can be observed with both alignment systems without any major difference between them. The decrease in precision can be explained by non-relevant service operations in the result list due to matching concepts that are not similar although being flagged as similar by the alignment systems. An increase in heterogeneity by introducing synonyms has a major impact on the quality of the results. After the introduction of synonyms the discovery provides results with poorer quality even than the initial discovery without any alignment. An interesting observation can be made when observing the results after the second alteration (vowels removal). A distinct difference between the two systems can be seen. LILY does not compute sufficient similarities between the concepts, thus only operations with concepts of the first provider are found. FalconOA on the other hand computes valid as well as invalid mappings, resulting in a lower precision but also in a higher recall and higher F-measure.

6 Current State-of-the-Art

Numerous approaches and implementations in the area of semantic service discovery already exist. Examples with high rates of citation are the WSMO-MX [24], a hybrid matchmaker, and discovery within the WSMX [25] (WSMO Execution Environment). However, research focusing on integration of automatic alignments in the area of service discovery is rather scarce. Worth mentioning is the work by [26], who investigate service discovery in an environment with many coexisting ontologies. Their system uses OWL-S descriptions and service requests are, aided by an "Ontology-Mapping Generator", translated into requests fitting the different ontologies. However, only in- and outputs are considered during discovery and no critical evaluation of the quality of alignments is provided. [27] consider additional semantic information in a business process environment to discover services. The authors do not explicitly develop a solution that is able to deal with heterogeneous ontologies. In order to allow heterogeneous service description the WSMO specification references the use of mediators, which are defined in an abstract way. Yet, additional information on how to generate or implement these mediators is not given [28]. One way to implement such mediators can be seen in *Glue* [29]. Nevertheless, the creation of these mediators is done manually, using domain and SWS experts.

7 Conclusions

This paper investigated the possibilities of supporting semantic service discovery using interchangeable ontology alignment systems. The approach is motivated by two facts: (1) the assumption that service providers will describe their services with concepts of a single homogeneous ontology is not realistic in a real world scenario, and

(2) a manual mapping between concepts of different service providers would lead to non-acceptable effort, especially with the multitude of service providers that are predicted to host services. In the beginning we clarified the fundamental terms and definitions for semantic service descriptions and their discovery. Additionally, the concepts of ontology alignment and ontology mapping were introduced. Furthermore, the proposed framework and its implementation were described. Based on the implementation it was possible to conduct an evaluation of service discovery without and with automatic alignments, including the alignment of ontologies with different degrees of heterogeneity.

We observed that using automatic alignments leads to a considerable increase in F-measure, considering a conservative amount of heterogeneity. This increase in result quality is mainly influenced by identifying additional hits in the services of the second provider, thus increasing the recall. However, false positive hits decrease the precision. Additionally, the cascading of errors from the alignment into the service discovery processes can be identified as a source for a lower overall F-measure. While this can be mainly observed in the case of the introduction of synonyms, it should be mentioned that the alignment systems used are not designed to directly accommodate synonyms. Considering this, future evaluations can be made using tools that use special techniques for synonym detection, such as the connection to a semantic net. Another finding is that the usage of realistic ontologies, as have been used in this scenario, leads to results inferior to those of the OAEI benchmarks. This may be traced back to the fact that the predefined benchmarks provide more and a higher quality of structural and semantic information which may not exist in all real world scenarios.

The presented approach can be generalized, meaning that although the current solution works with WSMO descriptions, other semantic service descriptions and their discovery may profit from automatic alignments. Most favorable should be descriptions in OWL, since the majority of current alignment systems are able to deal with OWL already.

Future research should be concerned with the evaluation of additional alignment systems. It is also desirable to increase the number of service requests and descriptions thus building a solid test collection. Furthermore, it would be interesting to see how the system could be enhanced to deal with requests on services of more than two service providers. This would introduce new requirements and strategies for the construction of alignments. Last but not least semi-automatic mappings should be considered as a trade-off between fully automated and manually constructed alignments. Manual intervention could be triggered once certain thresholds of similarity are reached, thus reducing the manual effort while allowing the identification of similar concepts that may not be computed easily (e.g. synonyms).

References

1. Offermann, P., Schröpfer, C., Holschke, O., Schönherr, M.: SOA: The IT-Architecture behind Service-Orientation. In: Steffens, U., Addicks, J.S., Streekmann, N. (eds.) MDD, SOA und IT-Management, pp. 1–11. GITO, Berlin (2007)
2. Lausen, H., Lara, R., Polleres, A., de Bruijn, J., Roman, D.: Semantic Annotation for Web Services. In: Rudi Studer, S.G., Abecker, A. (eds.) Semantic Web Services, pp. 179–210. Springer, Heidelberg (2007)

3. Berners-Lee, T., Hendler, J., Lassila, O.: The semantic Web. Scientific American 284, 28–37 (2001)
4. McGuinness, D.L., Harmelen, F.v.: OWL Web Ontology Language Overview (2004)
5. Martin, D., Burstein, M., McDermott, D., McIlraith, S., Paolucci, M., Sycara, K., McGuinness, D.L., Sirin, E., Srinivasan, N.: Bringing Semantics to Web Services with OWL-S. In: World Wide Web, vol. 10, pp. 243–277 (2007)
6. Roman, D., Keller, U., Lausen, H., Bruijn, J.d., Lara, R., Stollberg, M., Polleres, A., Feier, C., Bussler, C., Fensel, D.: Web Service Modeling Ontology. Applied Ontology 1, 77–106 (2005)
7. Keller, U., Lara, R., Lausen, H., Fensel, D.: Semantic Web Service Discovery in the WSMO Framework. In: Cardoso, J. (ed.) Semantic Web Services: Theory, Tools and Applications, pp. 281–316. Springer, Heidelberg (2007)
8. Akkiraju, R., Farell, J., Miller, J.A., Nagarajan, M., Sheth, A., Verma, K.: Web Service Semantics - WSDL-S (2005)
9. Farrell, J., Lausen, H.: Semantic Annotations for WSDL and XML Schema (2007)
10. Kopecký, J., Vitvar, T., Bournez, C., Farrell, J.: SAWSDL: Semantic Annotations for WSDL and XML Schema. IEEE Internet Computing, 60–67 (2007)
11. Grimm, S.: Discovery-Identifying Relevant Services. In: Studer, R., Grimm, S., Abecker, A. (eds.) Semantic Web Services. Springer, Heidelberg (2007)
12. Stollberg, M., Cimpian, E., Mocan, A., Fensel, D.: A Semantic Web Mediation Architecture. In: Koné, M.T., Lemire, D. (eds.) Canadian Semantic Web. Springer, US (2006)
13. Lara, R., Roman, D., Polleres, A., Fensel, D.: A Conceptual Comparison of WSMO and OWL-S. In: Zhang, L.-J., Jeckle, M. (eds.) ECOWS 2004. LNCS, vol. 3250, pp. 254–269. Springer, Heidelberg (2004)
14. Ehrig, M.: Ontology alignment: bridging the semantic gap. Springer, New York (2007)
15. Euzenat, J., Shvaiko, P.: Ontology Matching. Springer, New York (2007)
16. Hu, W., Qu, Y.: Falcon-AO: A practical ontology matching system. Web Semantics: Science, Services and Agents on the World Wide Web (2008)
17. Tang, J., Li, J., Liang, B., Huang, X., Li, Y., Wang, K.: Using Bayesian decision for ontology mapping. Web Semantics: Science, Services and Agents on the World Wide Web 4, 243–262 (2006)
18. Wang, P., Xu, B.: LILY: The Results for the Ontology Alignment Contest OAEI 2007. In: Proceedings of the International Workshop on Ontology Matching (OM) at the International Semantic Web Conference (ISWC)+ Asian Semantic Web Conference (ASWC), pp. 170–178 (2007)
19. Euzenat, J., Isaac, A., Meilicke, C., Shvaiko, P., Stuckenschmidt, H., Šváb, O., Svátek, V., van Hage, W.R., Yatskevich, M.: Results of the Ontology Alignment Evaluation Initiative 2007. In: Proceedings of the International Workshop on Ontology Matching (OM) at the International Semantic Web Conference (ISWC) + Asian Semantic Web Conference (ASWC), pp. 96–132 (2007)
20. Euzenat, J.: An API for ontology alignment. In: McIlraith, S.A., Plexousakis, D., van Harmelen, F. (eds.) ISWC 2004. LNCS, vol. 3298, pp. 698–712. Springer, Heidelberg (2004)
21. Sirin, E., Parsia, B., Grau, B.C., Kalyanpur, A., Katz, Y.: Pellet: A practical OWL-DL reasoner. Web Semantics: Science, Services and Agents on the World Wide Web 5, 51–53 (2007)
22. White, S.A.: Introduction to BPMN. IBM Cooperation (2004)
23. Sinkkila, R., Makela, E., Kauppinen, T., Hyvonen, E.: Combining Context Navigation with Semantic Autocompletion to Solve Problems in Concept Selection. In: First International Workshop on Semantic Metadata Management and Applications, SeMMA 2008, Located at the Fifth European Semantic Web Conference (ESWC 2008), vol. 346, pp. 61–68 (2008)

24. Kaufer, F., Klusch, M.: WSMO-MX: a logic programming based hybrid service matchmaker. In: Proceedings of the 4th IEEE European Conference on Web Services (ECOWS 2006), pp. 161–170 (2006)
25. Haller, A., Cimpian, E., Mocan, A., Oren, E., Bussler, C.: WSMX - a semantic service-oriented architecture. In: Proceedings of the 2005 IEEE International Conference on Web Services (ICWS), vol. 1, pp. 321–328 (2005)
26. Liu, Z., Wang, H., Zhou, B.: A scalable mechanism of semantic service discovery in multi-ontology environment. In: Cérin, C., Li, K.-C. (eds.) GPC 2007. LNCS, vol. 4459, pp. 136–145. Springer, Heidelberg (2007)
27. May, N., Weber, I.: Information Gathering for Semantic Service Discovery and Composition in Business Process Modeling. In: Dietz, J.L.G., et al. (eds.) CIAO! 2008 and EOMAS 2008. LNBIP, vol. 10, pp. 46–60. Springer, Heidelberg (2008)
28. Fensel, D., Lausen, H., Polleres, A., de Bruijn, J., Stollberg, M., Roman, D., Domingue, J.: Enabling Semantic Web Services: The Web Service Modeling Ontology. Springer, Berlin (2006)
29. Della Valle, E., Cerizza, D.: The mediators centric approach to automatic web service discovery of glue. In: Hepp, M., Polleres, A., van Harmelen, F., Genesereth, M. (eds.) Proceedings of the First International Workshop on Mediation in Semantic Web Services (MEDIATE 2005), vol. 168, pp. 35–50 (2005)

Towards an Implementation of the EU Services Directive with Semantic Web Services

Ingo Weber and York Sure

SAP Research, CEC Karlsruhe, Germany
{ingo.weber,york.sure}@sap.com

Abstract. The EU Services Directive aims at easing the burdens for the EU's citizens to open up new businesses by providing a single-point-of-contact for the complete business lifecycle. It has to be implemented by all EU member states by the end of 2009. The key technical challenge is the strong dependency between the individual situation of each citizen and the underlying business process: the resulting large variability of possible processes makes it difficult to pre-configure a system for dealing with all required variants. To overcome this bottleneck we present a method and a prototype for automatic service composition, visualization, monitoring, and execution based on Semantic Web Services.

1 Introduction

1.1 Motivating Scenario

Assume that Pablo Benitez, a Spanish citizen, wants to move to the German city Cologne to open up a café with the possibility to sell liquor. In order to get permission to open the café, he has to register as a resident in Cologne; obtain proofs of his record from the police, the Registry of Trade and Industrial Offences, and the tax office; register with the commercial registry; obtain a tax number and tax clearance for the new business; get the liquor license from the office of business affairs; and finally register his new business. Many of these steps require him to fill in a number of forms, giving away – among other information – his current, new, and business addresses, information about his marriage, both of his parents, whether he participated or signed up for an integration course, and so on. Most of the forms today are only available in German; and while his German language skills may be sufficient to operate a café and to interact with customers, the language used in public administration may use an entirely different vocabulary. Further, he is likely to fill in the same data into many different forms, e.g., his addresses. Eventually, the forms need to be signed and delivered to the various offices, sometimes he has to show up personally. Finding out the steps and the dependencies of the business process in first place is a difficulty in itself as each opening of a business is a highly individual process that is dependent on the individual situation of the founder, the kind of business and the location of the business. Clearly, these circumstances make Pablo think twice, before making the step to open up the café; and may even make him regret this decision various times before the actual opening.

W. Abramowicz (Ed.): BIS 2009, LNBIP 21, pp. 217–227, 2009.

1.2 The EU Services Directive and Related Work

Having understood these hardships and the negative economic aspects and hindrance for a more tightly integrated Europe, the European Union decided to improve the situation by means of the so-called "EU Services Directive" [1]. This directive requires all public entities at the level of a city council or "Gemeinde" to offer their services, at least in part, over the internet. In the above example, Pablo has to have the opportunity to request the licenses irrespective of his current location, while being able to monitor the progress of the licensing process remotely.

Key requirement of the EU Services Directive is to establish a Single-Point-of-Contact (SPoC) that serves as the sole interface to the citizen and coordinates the execution of the application process – and later on for the complete lifecycle of a business. The law does not prescribe how such a SPoC has to be provided, be it via internet or via a call center, nor on which granularity it has to be established, may it e.g. be on a federal state level or on a city level. These issues are still under discussion in the EU member states.

For this paper we assume the implementation as a Web-based solution. The level of granularity (federal state or city or …) can in principle be neglected, however, we assume that there will be exactly one or a few SPoCs for e.g. each of the German federal states. We further assume that SPoC interacts with the applicant via the Web and provides a Web-based questionnaire as the user interface to the applicant in which all necessary data for handling the licensing request is collected. Although the SPoC eventually should cover the full lifecycle of businesses, we concentrate in this paper on the very beginning of the process, i.e., the opening of a new business.

In the following we do not claim to present an exhaustive list of requirements, but we concentrate on the most relevant ones: the key requirements are to (i) minimize the burden for citizens, (ii) guarantee that a certain time limit for the overall process of opening of a new business is not exceeded, and (iii) ensure the continuous monitoring of the process execution for the applicant.

To lower the burden for the citizen all required data should only be requested once. The individual application process instance has to be created, executed and monitored based on the collected data, in particular the data about the situation and the desired business registration, which essentially means that the SPoC has to know which kinds of permits have to be requested for a particular case and which concrete public authority or other institution is responsible for handling the request and granting the permit.

To minimize the duration for each application the process has to be optimized during the planning, i.e. process steps have to be parallelized as much as possible. This means that while one permit is being processed, another one which is independent of the first one can be processed at the same time. Essentially this means that the SPoC has to have knowledge about the dependencies of all process steps.

Several thousand public authorities are involved in handling licensing requests, and their involvement differs depending on the particular license – e.g., the processes for opening a café or a barber's shop differ substantially. Further, all these authorities have the legal power to design how they offer particular services. As argued above, the situations in which a citizen may be can vary substantially – e.g., for opening a barber's shop, suitable education is required; but which of the many educational offerings in today's 27 member states are perceived as suitable is a non-trivial question.

Thus, we assume that dealing with all these variations by manually designing all process variants or by means of configurable process models [10][11] is infeasible. Instead, a dynamic composition of the process is required where the individual situation of the applicant is mapped to the services from the various services providers, i.e. for Germany the local authorities. As we will see later, our approach makes use of knowing the input and output dependencies of each individual process step to compute an optimal composition into the overall process including a suitable parallelization of process steps.

As the SPoC also has to handle many requests at the same time, the composition of the individual process instances has to be automated. This allows the SPoC to concentrate on monitoring process execution and handle exceptions. This is important as a particular process instance has to be executed within a given time frame. Should the processing time exceed this threshold, then the request must be approved automatically. To avoid such cases, it is of necessity to equip the SPoC with monitoring options for the process execution.

Another challenging point is the question how the SPoC can be enabled to collect and maintain the knowledge about all authorities involved and their service offerings. Due to the high number of authorities and services, the most pragmatic solution is to collect and maintain the information in a decentralized way. A detailed discussion of this point is subject to our ongoing work, but out of scope for this paper.

In [1] a framework architecture for the realization of a single-point-of-contact in Germany is described. The architecture is very high level, as they want to allow for different specializations of this architecture e.g. in the different German federal states. In contrast, this paper describes one particular specialization of this framework architecture. The main difference lies in the way how processes that are executed by the SPoC are generated. In the related work they are pre-defined and can be adapted by ad-hoc workflows, if necessary for a specific case. In contrast to that we dynamically compose the processes making use of the Central Services and Administration Registries.

In general, our approach relies on the use of Semantic Web Services. A good overview of recent research is given in [8]. More specifically we compose services automatically based on goals, pre- and post-conditions such as described by the framework around the Web Service Modeling Ontology (WSMO) [9]. However, we herein pursue a systems approach, i.e., we make use of existing components where possible, and describe how the missing pieces can be designed. For instance, we use the composition approach described in [3], but do not make technical contributions to the composition itself. For a treatment of related work on other composition approaches, the interested reader is thus referred to [3].

Since central coordination is of high importance, we pursue an orchestration approach and do not consider autonomous agent technology [13].

1.3 Solution Approach

The desired SPoC collects the relevant data from the applicant only once for all process steps where it is required via a Web form which adapts on-the-fly according to entries of the applicant; on this basis the SPoC composes the optimal licensing process and computes its costs and duration; then the SPoC executes the process without intermediate delays, given all process steps perform as expected and all licenses are granted. Since all data is handled and forwarded by information systems, media disruptions can be avoided and the process execution time may be lowered considerably. In general, through the centralized view on the process, transparency in terms of operational relations and effort in terms of time and cost is increased.

In the next section we describe the main contribution of this paper: our overall approach for an implementation of the EU Services Directive. Section 3 discusses the proposed approach and concludes the paper.

2 Implementing the EU Services Directive

In this section we describe our solution approach along the lines of the behavioral view depicted in Fig. 1.

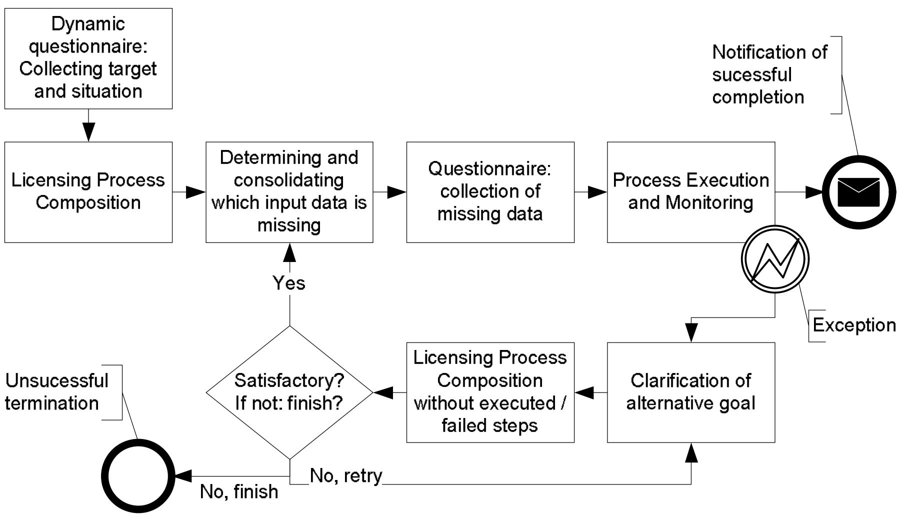


Fig. 1. Behavioral view of our solution approach

We give details on how the situation of the citizen is captured by dynamic questionnaires (Section 2.1), which enables the automatic composition (Section 2.2). The composed process then is displayed to the user (Section 2.3). From the process we can infer which information is required and not yet available, and the missing data can be requested from the user (Section 2.4). Once the data entry is completed, the process can be executed and monitored during execution by the stakeholders (Section 2.5). How to deal with exceptions during the execution is described in Section 2.6.

The descriptions of the behavioral view make use of components which are shown in the structural view of the proposed system, Fig. 2.

A user, be it a human agent of the SPoC or the applicant himself, interacts with the Web server of the SPoC through a Web browser for the purpose of requesting the license of his choice. This Web server provides two kinds of questionnaires: (i) the Dynamic Goal Questionnaire, which is used to enter the license type, its conditions, and the situation of the requestor; and (ii) Data Questionnaire for entering the missing data for the licensing steps. The former part is used for the data for the semantic goal, and the latter (potentially together with parts of the former) is the actual input data for the individual services. In the following, details of the individual steps are provided.

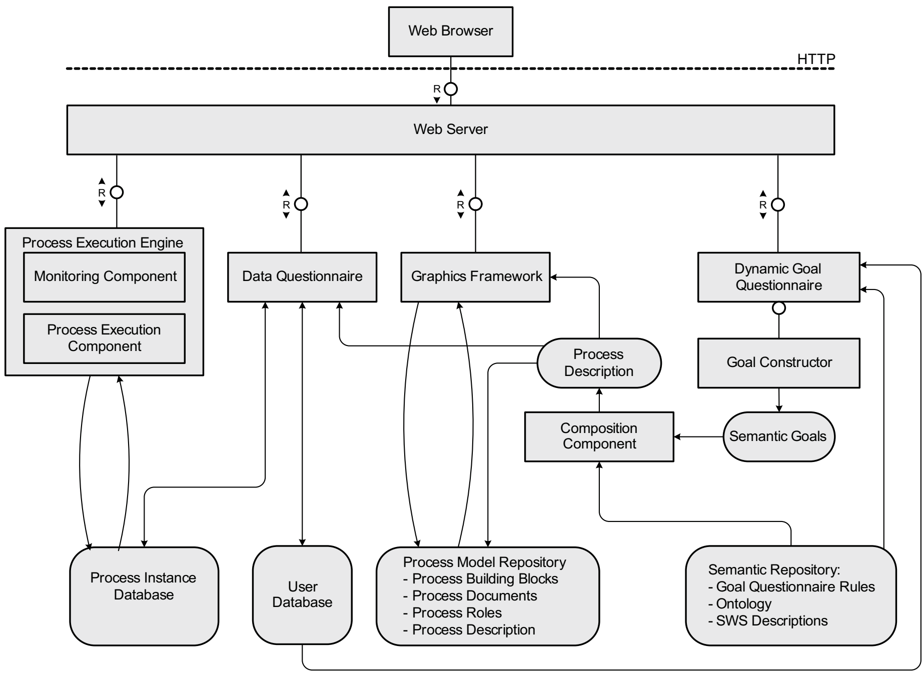


Fig. 2. Structural view of the suggested system (in FMC Component-Block notation[1])

2.1 Dynamic Questionnaire: Collecting Target and Situation

As a first step, the *Dynamic Goal Questionnaire* is filled in. Based on the *Rules for Questionnaire Creation*, the information that is relevant to the goal is requested from the user. After selecting the main license type (e.g., opening a Café in Cologne), these rules guide which information is relevant to the main license type, (e.g.: is the requestor interested in hosting life music performances or offering a food menu?). As this is an iterative procedure of data entering and rule evaluation we call it "dynamic".

Once the goal is specified, the situation of the applicant needs to be entered (e.g., a Spanish citizen with education certificates from the Netherlands and an Italian wife). Parts of this information may already be available in the system, namely in the *User Database*. The question which information about the situation of the citizen is relevant for the licensing depends on the desired license. In fact, the data that may legally be collected by authorities may be restricted, again with respect to the desired license.

[1] http://www.fmc-modeling.org/

2.2 Licensing Process Composition

The input gained from the *Dynamic Goal Questionnaire* is taken as input to the *Goal Constructor*, which creates the semantic goal (i.e., according to WSMO) for this particular requestor. This input contains the information on the current situation of the requestor and the desired license including the license's relevant parameters. Abstractly speaking, the task of the automatic composition is to find a sequence of (atomic) services which, starting from the current situation of the requestor gets him to the target state of having the given license. The services here correspond to the licensing steps. The input information from the questionnaire is interpreted by the Goal Constructor, which constructs logical state representations of the current and the targeted state. In composition parlance, these states constitute the precondition (the current state as the situation of the requestor) and the postcondition (the target state) of the goal.[2]

With this goal, the *Composition Component* is invoked. As said, the task of the Composition Component is to find a sequence (or, more generally, an orchestration) of services which together can fulfill the goal. As the goal, the services are described in terms of their preconditions and postconditions (also referred to as effects). Note that a composition based on inputs / outputs only does in general *not* suffice to solve the problem at hand. This is due to the fact that the semantics of a service cannot be captured in inputs / outputs only[3]. Any Composition Component which can compose atomic services based on their preconditions and effects can in principle be used here. Examples for works on automatic composition of semantic Web services include [3][4][5], and are often based on AI Planning results, such as [6].

In particular, our prototype uses the implementation presented in [3]. In short, this composition approach builds on the FF algorithm [6] from AI planning, and extends it with specifics for SWS composition. It performs a forward search in the space of possible compositions. This search is guided by a suitable heuristic function, i.e., computing solutions to a relaxed composition problem which is exponentially easier than the full problem. Further, an ontology (with limited expressivity) can be taken into account. A purely sequential solution is composed at first, and parallelized as far as possible in a post-processing step. Due to the error handling procedure discussed in Section 2.6, there is no need for composing a contingency plan containing case distinctions for possible deviations from the desired outcomes of the process steps. The composition approach in [3] is particularly well suited for the scenario presented in this paper, since it scales up well to large service repositories. However, other approaches could in principle be used too.

Regardless of the specific composition approach used, the Composition Component takes the available *Semantic Web Service (SWS) Descriptions* and the *Ontology*[4] from the *Semantic Repository*, and automatically composes a *Process Description* for the requestor. The *SWS Descriptions* are abstract descriptions of the Process Building

[2] Note that the precondition is perceived as part of the composition goal.

[3] The unconvinced reader may consider a service with a car description as input and a price as output. Does the service buy cars, sell cars, or assess the value of cars?

[4] The ontology serves as a model for vocabulary and relations in the domain of discourse. The approach can also make do if the relations are not modeled (or empty), or if the vocabulary is presented in some other form. For our solution we have engineered an appropriate ontology together with domain experts from the SAP Public Sector field organization.

Blocks which reside in the *Process Repository*. In SWS parlance, the *SWS Description* contains a "grounding" to a *Process Building Block*. Note that the *Process Building Blocks* are not required to be implemented as Web services, but we describe them using SWS standards. Note further, that the differentiation between the two repositories in the architecture is a logical one, i.e., they may be implemented as a single system; this also impacts the pragmatics of life-cycle management of the *Process Building Blocks*, which again is outside of the scope of this paper.

There is a notable difference to standard workflow approaches, where a model is created once and executed in many instances; here, the model is created *for* the instance and subsequently executed.

The *Process Building Blocks* may specify the following additional information (i.e., in addition to their precondition and effect):

- The required input data, e.g., name, maiden name, address, date of birth, passport number,
- the actors who are involved in performing the activity, e.g., the central federal registry of Germany (de: "Bundeszentralregister") is responsible for creating police record excerpts,
- the maximum duration, e.g., two days, and
- the price, e.g., 10€.

In order to give overview information on the process level to the user, the respective sums of the latter two aspects are computed. Since a shorter duration of the licensing process is desired, the composed process is (ideally) parallelized to a maximum degree – i.e.: if a step in the process is not dependent on some other step (directly or indirectly), or vice versa, our system assumes that those two steps can be scheduled for parallel execution. Depending on the exact execution semantics of the process execution language, the maximal process duration may be calculated as the sum of durations of the longest sequence of contiguous steps in the process. The sum of the cost can be computed in an easier way: it is simply the sum of all costs of *Process Building Blocks* in the process.

A screenshot of the composed process is shown in Fig. 3 where each step of the process is shown as a blue box, starting from the top going to the bottom where parallel processes do occur (to minimize duration of the overall process). For the step *Assign Tax Number* a yellow box, which appears with a mouse over, shows the annotated meta data including the actor, the pre- and post-condition (internally represented according to WSMO), the time and the costs.

2.3 Presentation of the Composed Process to the User

The composed process then is persisted in the *Process Model Repository* for later use, e.g., through the *Process Execution Environment*[5]. Next, the *Process Description* is displayed to the user on a Web site, e.g., in the form depicted in Fig. 3, by using a *Graphics Framework*.

[5] A standard WSBPEL [12]execution engine or a workflow execution engine can be used as execution environment. If the format in which the process model is represented is not interpretable by the execution engine at hand, a translation needs to take place. In this paper we neglect the details of the format, since the specifics have no impact on our approach.

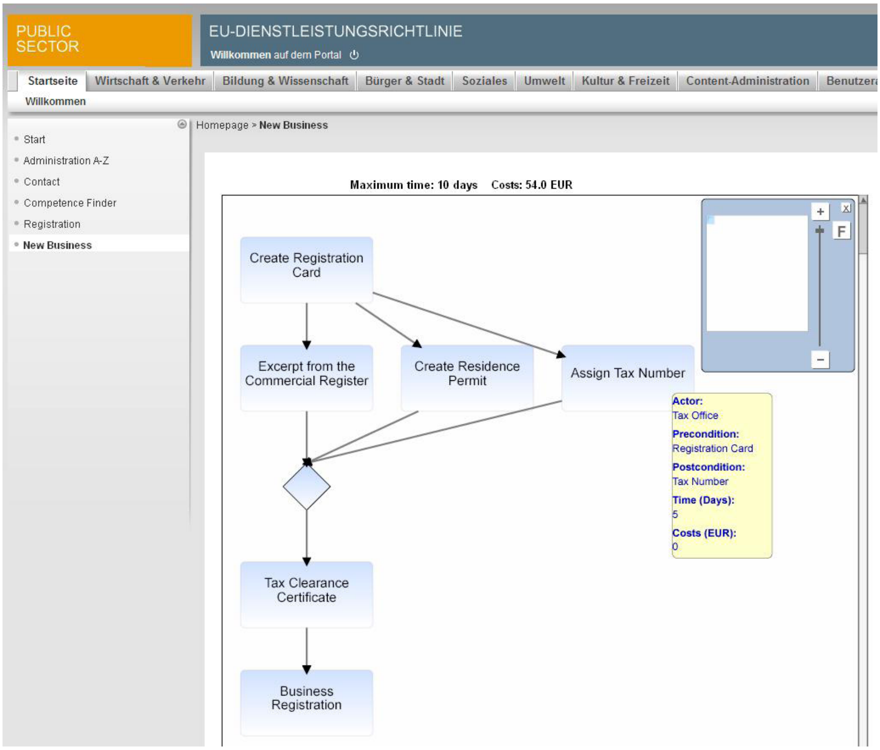


Fig. 3. Screenshot of the prototypical implementation: automatically composed process for opening of a Café of a Spanish citizen in Germany

Note that this view may only be appropriate if the user is a representative of the SPoC; if the user is the applicant, then this view may be hidden. Otherwise, the *Graphics Framework* automatically creates a layout for the composed process model and renders the graphical representation into a format that can be transmitted by the *Web server* and displayed in the *Web browser* of the user. Our prototype, e.g., uses support vector graphics (SVG) for this purpose.

2.4 Consolidation of Input Data and Collection of Missing Data

Once the *Process Description* has been rendered and graphically displayed to the user, he may select to start the execution of the process, in which case it first has to be clarified if all required input data is available. This is done by iterating over the composed process steps and collecting the required input data for all of them. This collection then is consolidated, i.e., it is checked if the same data is used by multiple steps (e.g., name, address of the requested pub, home address). The consolidation can be based on data standards, e.g. UN/CEFACT CCTS [7]. Subsequently, it can be compared (using the same technique) whether the consolidated input data is already available in the system (in the *User Database*), or whether it still has to be entered.

If there is missing data, it is requested through the *Data Questionnaire* from the user. The data gained here can be stored in the *User Database* - if legally permissible and agreed to by the user. As for the licensing process, the goal is to collect all necessary data upfront, once, and to progress the licensing process afterwards without intermediate delays. This feature by itself is already quite a significant benefit for the requestor, as it speeds up the process and massively lowers the burden of filling in forms in comparison to today.[6]

2.5 Process Execution and Monitoring

Finally, when the process model and all input data are available, the process execution can be started. The execution is mostly steered by the *Process Execution Environment* (cf. footnote 5). The progress made can be reviewed through a *Monitoring Component*, while the actual process control remains with the *Process Execution Component*. During the execution, *Process Instance Data* is persisted in the *Instance Database*, also for purposes of potential auditing later on.

2.6 Exception Handling

Since many of the process steps are requests for granting certain permissions or the like, there is a chance that not all of them complete successfully – e.g., the permission for live music performances may not be granted for locations in quiet residential areas. For such exceptions, the system allows to *re-plan* the process, given the new information. The requestor is pro-actively notified (e.g., via e-mail) of the changed situation and may choose an alternative goal; the options are presented through the *Dynamic Goal Questionnaire* again.

If an alternative goal is chosen, the *Composition Component* is invoked with a new goal from the *Goal Constructor* together with the current state of the process. If a satisfactory solution can be found this way, the procedure is restarted at the point where the input data is collected. Additional data that was not required before is collected from the requestor at this stage. In contrast, if no satisfactory solution can be found for the user, even with multiple attempts, the process is terminated in an unsuccessful state.

In contrast, if all licenses are granted or occurring exceptions are handled satisfactorily, the applicant is informed about the successful outcome of his request. He now may start doing a hopefully successful business in Germany.

3 Conclusions and Outlook

In this paper we presented a conceptual architecture and a running prototype for implementing the key requirements imposed by "EU Services Directive" to the European member states. In a nutshell, these requirements are: to offer a Single-Point-of-Contact for distributed public services; the possibilities to intelligently create license processes for citizens; to pro-actively inform citizens about required changes in the process due to newly available information; and to allow citizens to remotely monitor the status of their inquiries.

[6] Anecdotic evidence tells us that more than one hundred forms may have to be filled in today in order to open a barber's shop in Germany today (given the requestor is a non-German EU citizen who does not live in Germany yet). Unfortunately we could not verify the anecdote so far due to time constraints.

At the core of the solution is a composition component which can, based on a semantic goal, create a licensing process tailored to the situation and objectives of an individual requestor. In order to enable the use of this approach, a dynamic questionnaire collects the relevant information on the current situation of the requestor as well as of the specific conditions of the requested license. Subsequently the automatic composition of previously specified process steps, or services, can be performed. Once the process has been composed, the system checks if the input data for all licensing steps in the process is available, so that the process can be executed without intermediate delays. If that is not the case, the missing input data is collected. When all data is available, the execution is started.

Due to the nature of the process steps it may well be that some of them deliver a negative outcome, e.g., if a license is not granted. In this case, our approach helps in dynamically re-planning for the changed situation, e.g., suggesting alternatives to some desired license conditions or suggesting a different (potentially more pricey) way to achieve the original goal.

Based on the upcoming necessity to automatically grant licenses as soon as the processing time exceeds the threshold, automatic monitoring seems to be a desirable feature for the SPoC. In the future, this may be implemented on the basis of the solution presented here. Further, the SPoC provides most benefit to all EU citizens if the data entries may be made available in multiple languages – be it through human agents working for the SPoC, or through forms on Web sites, where the effort of translation has to be spent only once. However, this also has been out-of-scope for the initial prototype but will be considered in later versions.

The EU member states who are leading in providing their citizens electronic access to public services usually “only” offer access to electronic forms which can be printed and filled in at home, along with descriptions of public processes to guide citizens. The next wave will be the provisioning of all kinds of public services as electronic services, including all services that are needed to implement the EU Services Directive.

While the presented approach is designed for the particular setting of the "EU Services Directive", we see it as the blueprint for various scenarios of composite citizen-centric services. First discussions with colleagues from the Public Services field organisation at SAP indicate that our solution is broad enough to support many different kinds of public services – but this still has to be proven. Key questions for future research are: how to ensure completeness of the created processes; how to achieve required coverage of public sector processes; evaluate the feasibility of semantic service annotation and consider automated approaches for service annotation; consider security aspects for public services.

References

[1] Further information on the EU Services Directive (observed on 2009-02-02), `http://ec.europa.eu/internal_market/services/services-dir/index_en.htm`

[2] Lucke, J. v., Eckert, K.P., Breitenstrom, C.: EU-DLR - Einheitlicher Ansprechpartner: Rahmenarchitektur und technischer Lösungsvorschlag. Whitepaper, Fraunhofer-Institut für Offene Kommunikationssysteme (FOKUS) (February 2008)

[3] Hoffmann, J., Weber, I., Scicluna, J., Kaczmarek, T., Ankolekar, A.: Combining Scalability and Expressivity in the Automatic Composition of Semantic Web Services. In: ICWE 2008: Intl. Conf. on Web Engineering (July 2008)
[4] Meyer, H., Weske, M.: Automated service composition using heuristic search. In: Dustdar, S., Fiadeiro, J.L., Sheth, A.P. (eds.) BPM 2006. LNCS, vol. 4102, pp. 81–96. Springer, Heidelberg (2006)
[5] Constantinescu, I., Faltings, B., Binder, W.: Large Scale, Type-compatible Service Composition. In: International Conference on Web Services (2004)
[6] Hoffmann, J., Nebel, B.: The FF Planning System: Fast Plan Generation Through Heuristic Search. Journal of Artificial Intelligence Research (2001)
[7] United Nations Centre for Trade Facilitation and Electronic Business (UN/CEFACT). Core Components Technical Specification. Part 8 of the ebXML Framework (November 2003), http://www.unece.org/cefact/ebxml/CCTS_V2-01_Final.pdf
[8] Studer, R., Grimm, S., Abecker, A. (eds.): Semantic Web Services - Concepts, Technologies, and Applications. Springer, Heidelberg (2007)
[9] Bruijn, J.d., Fensel, D., Keller, U., Lara, R.: Using the Web Service Modeling Ontology to Enable Semantic E-Business. Communications of the ACM 48(12), 43–47 (2005)
[10] Gottschalk, F., Van Der Aalst, W.M.P., Jansen-Vullers, M.H., La Rosa, M.: Configurable Workflow Models. International Journal of Cooperative Information Systems (IJCIS) (2008)
[11] Rosemann, M., Van Der Aalst, W.M.P.: A Configurable Reference Modeling Language, Information Systems. Elsevier, Amsterdam (2007)
[12] Alves, A., et al.: Web Services Business Process Execution Language version 2.0, Committee specification, OASIS (January 2007)
[13] Russel, S., Norvig, P.: Artificial Intelligence: A Modern Approach, 2nd edn. Prentice-Hall, Englewood Cliffs (2002)

Integrating Process and Data Models to Aid Configuration of ERP Packages

Tripti Negi and Veena Bansal

Department of Industrial and Management Engineering
Indian Institute of Technology, Kanpur, 208016, India
{negi,veena}@iitk.ac.in

Abstract. Configuration of ERP systems is complex. Configuration uses output of Requirements Engineering (RE) phase to select relevant tables, parameters and set relationship among tables of the ERP database. Here, we prove that a gap exists between the output of RE phase and input of configuration phase. The reason for this gap is that configuration looks at a process from data perspective (data elements, their interactions and interrelationships) whereas RE techniques view a process from an activity flow perspective. To minimize this gap, data perspective should be captured in RE phase. Here, a set of required information for configuration is identified /defined. We propose to use a data modeling technique along with process modeling technique in RE phase to support configuration specific information. Widely used existing data modeling techniques do not capture data specific configuration requirements. Thus, a data modeling technique called Data Activity Model for Configuration (DAMC) is proposed.

Keywords: Configuration, Data model, ERP, Requirements Engineering.

1 Introduction

ERP systems are software packages that integrate business processes of an organization through data/information flow [9] and requires configuration. Configuration is done at two levels: at system level and at business process level [1]. Appropriate hardware and software including database system, network system and operating system are selected and configured as a part of system configuration. Business process configuration (BPC) configures ERP solution to capture business processes of the client organization. A business process is configured by selecting appropriate tables, attributes and relationships between these tables. The discussion for configuration in this paper refers to BPC.

Configuration is done using models and documents produced in RE phase. The assumption is that the information required during configuration phase will be captured during RE phase. There are four aspects of a business that translate into requirements namely data/ information, processes, business rules and external requirements. The objective of the model created in RE phase is to capture all four business aspects. A research study [11] shows that RE techniques, as Object Process Methodology and

W. Abramowicz (Ed.): BIS 2009, LNBIP 21, pp. 228–239, 2009.

Event-driven Process Chain (EPC) diagram are process-oriented techniques that lay emphasis on functions, processes their interaction and the required data. However, during configuration, the emphasis shifts to data components, their transformation and interactions that accomplish a function or task. These RE techniques completely do not capture data interaction and data transformation to accomplish a function. Thus, there is a considerable gap between the information available and the information required for the configuration phase. This gap results in extensive configuration efforts that may eventually lead to significant implementation cost and failure.

To minimize or close this gap, there is a need to create data model along with process model in RE phase. The data modeling technique along with a process modeling technique together will capture information from real world (organization) and will provide reference for ERP configuration as shown in figure 1.

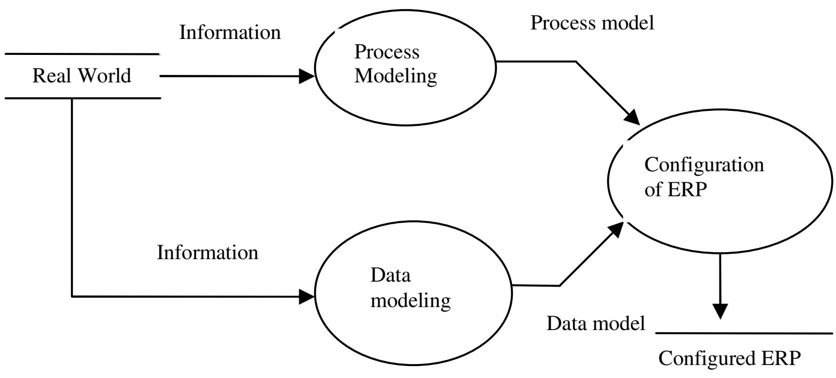


Fig. 1. Importance of integrating process and data models for Configuration

Along with this, RE phase should capture and model information needed for configuration.

In this paper, information essential for configuration is identified /defined in section 2. We take example of Configurable–EPC (C-EPC) in section 3 to show a RE process modeling technique does not capture data specific configuration details. Few, existing data modeling techniques are analyzed in section 4 followed by proposing a new data modeling technique called DAMC (Data Activity Model for Configuration) in section 5 to capture data specific configuration requirements. Section 6 validates DAMC followed by conclusion in section 7.

2 Information Needed for Configuration

Configuration of ERP system involves comprehensive selection of relevant parameters from appropriate tables and activating relationships between tables from the backend database [14]. Enabling/disabling of the parameters/attributes in the tables map the corresponding business processes to the ERP system [8]. The number of tables and attributes to choose from may run into thousands.

The following information about data and processes are significant for configuration [13] [14]:

1. Information regarding processes, functions, control flow, and data is required for configuration. Process should be defined in terms of its functionality and control flow.
2. While configuring the ERP system, many decisions about alternative functions, tables, and parameters is to be made. Configuration reference models should be able to capture these decisions. The decisions can either be mandatory or optional decisions. Some configuration decisions are critical while others are non-critical. Critical decisions have significant impact on the business processes. The decisions should be differentiated between critical and non-critical decisions.
3. There should be a logical sequence to take decisions for configuration. A decision may include processes and data objects. The logical sequence is through interrelationships within one process model, between one or more process models or between process and data models.
4. ERP specific configuration details may also be required. For instance, SAP configuration is done through its IMG (implementation guide). Such information may provide valuable information to the configuration team.
5. Reference data models are important for configuration. ER diagram or its variants are used for data models. We need to distinguish between optional entities and required entities. An entity is connected to another entity through a relationship. A relationship may be optional. Distinction between mandatory and optional relationship is required.

Other than the above-mentioned features, we define following information important for representing data and its interaction.

6. Distinction between persistent (master) and transactional data is important. Transactional data is usually created when a transaction is executed because of successful configuration. It is master data that is used for configuration.
7. Representation of Business Rules: A business process is executed with respect to business rule/s. These business rules prescribe a certain action or constraint the set of possible actions [7]. Business processes have integrity rules and activity rules [7]. ERP database incorporate integrity rules as they define integrity constraints on databases. An integrity constraint is an assertion that must be satisfied in all evolving states. Hence, we do not consider them explicitly while modeling for configuration. Activity rules prescribe actions or operation sequences that are to be performed and incorporated in any process and/or information constraint that is required for an activity [7]. Activity rules can either be structural or operational in nature. The type of relationships and cardinality constraints define structural rules. Cardinality expresses maximum number of entities that can be associated with another through a relationship. Operational rules define pre-condition/s and post-condition/s for an activity. A precondition is a constraint that specifies what must be true before an activity is performed whereas post condition is a constraint that is true after an activity is accomplished [5]. While configuration,

we need to know what business rules accomplish an activity along with knowing the type and cardinalities of a relationship. Thus, structural and pre-condition activity rules are required while modeling.

3 Analysis of C-EPC

Configurable EPC (C-EPC) is an extension of EPC model to include configuration specific information while capturing requirements [13]. This information includes taking decisions about various alternatives available for a function and connector to be included while configuration.

Figure 2 shows a part of C-EPC corresponding *sales and order* process. The sales process starts by creating a *sales order* when *customer demand exists* (event). If an organization decides to relate customer demand to *sales order*, *create sales order* function is included else it is excluded or temporarily not considered while configuration. Thus, this function is a configurable function. Once *sales order* is created, availability of stock is checked. If stock does not exist, material is procured by creating *purchase order.* Procurement can be for goods or for services and it depends upon the type of organization resulting in XOR configurable connector.

C-EPC supports process specific configuration requirements by modeling mandatory and optional functions and connectors for a process. However, nothing is mentioned about the configuration requirements of data elements. There is no means to distinguish between required and optional data elements, persistent and transactional data, mandatory and optional relationship between data elements. It cannot represent business rules. This makes C-EPC unsuitable to depict data elements and their interaction for configuration purpose, although it is the only modeling technique that supports process specific configuration requirements.

Thus there is a need for a data modeling technique to be used along with an EPC model (EPC results as a variation of C-EPC) as shown in figure 3 to capture data specific configuration details. The next section analyses data modeling techniques.

4 Analysis of Data Modeling Techniques

A data model is created to depict data elements and their interactions for *create sales order* activity corresponding to EPC model of figure 3. *Create sales order* activity progresses as follows: first outstanding credit for a customer is checked. If credit is within limit, discount if applicable is associated with material and customer. Tax if applicable to a material is computed and finally sales order is created for a customer. The entities involved are *sales order, credit, customer, tax, discount* and *material.*

Create sales order activity is modeled using EER notations [4] as shown in figure 4. EER diagram has no means to distinguish between persistent and transactional data and between mandatory and optional relationship. Business rules that express constraints on a relationship are represented, but pre-condition business rule cannot be depicted using this method. For instance, *sales order* can only be created if a *customer* has valid *credit* limit. This rule is shown through arrow pointing towards relationship *associated with* and connected to the relationships *associated with* and has between

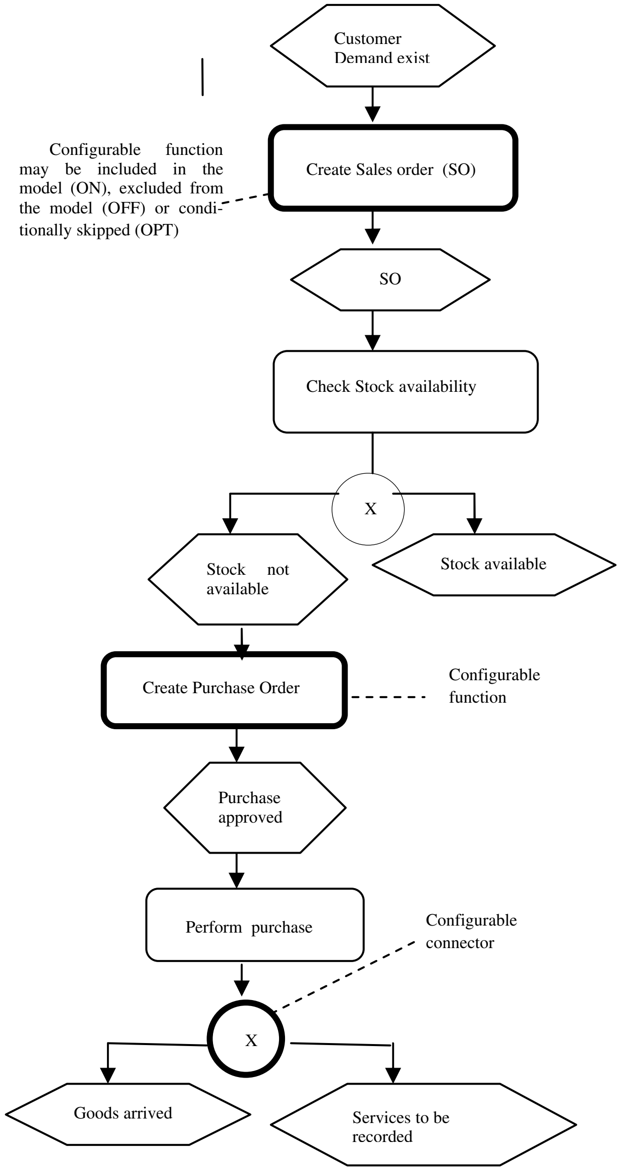


Fig. 2. C-EPC for a part of sales order process

entities *customer-sales order* and *customer-credit.* However, pre-condition rule that states that *sales order* can only be created if either the quotation is available or purchase order is available cannot be represented using this technique. Hence, EER model does not completely fulfill configuration specific requirements.

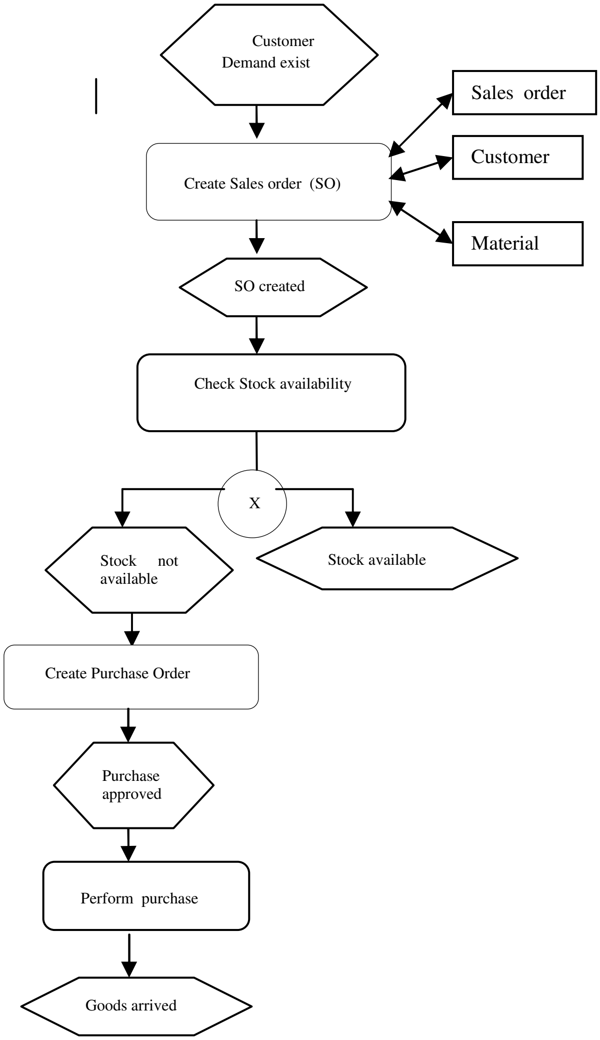


Fig. 3. Variation of C-EPC

Figure 5 depicts class diagram [2] corresponding to *create sales order* activity.

A unidirectional relationship between class *customer* and class *credit* is given by solid black triangle pointing towards the class *credit* indicating that a *customer-has-credit*. A ternary relationship between *customer, material* and *sales order* is depicted with a large open diamond. Class diagram does not fulfill configuration specific requirements like distinguishing between persistent and transactional data, between mandatory and optional relationship. Pre-condition business rules cannot be represented through class diagram.

Thus, in the next section we propose a data modeling technique that will capture configuration specific data requirements.

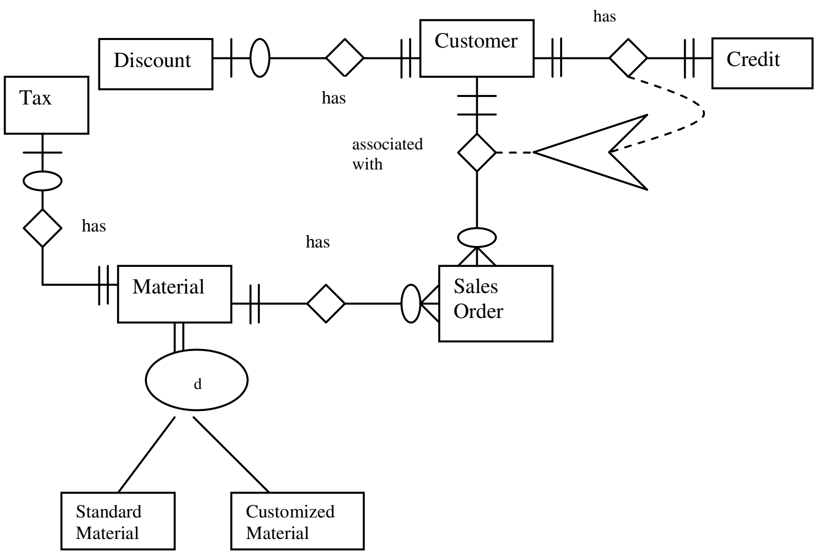


Fig. 4. EER diagram corresponding to Create Sales Order

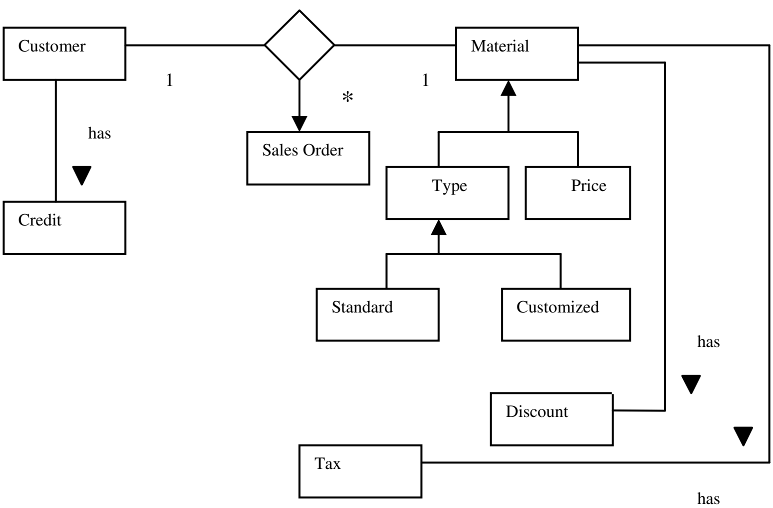


Fig. 5. Class diagram for Create Sales Order

5 Proposed Data Modeling Technique

We propose a data modeling technique called Data Activity Model for Configuration (DAMC) in this section.

5.1 DAMC – An Explanation

DAMC has notations to represent entity, its attributes and relationship between entities. An entity represents data. Data can be either master data or transactional data. Entity representing master data entity is depicted by a rectangle whereas double lined rectangle denotes transactional data entity. An entity has attributes. Ellipse denotes an attribute.

A relationship connects entities. A relationship can be either mandatory or optional in nature. A solid line shows mandatory relationship whereas a dotted line indicates an optional relationship. The entities participating in an optional relationship may themselves be optional. A mandatory (required) entity is represented with an *r* and an optional entity is depicted with an *o* written in the corner of an entity. The structural rules are depicted as cardinality constraints. One-to-one, one-to-many and many-to-many cardinality constraints are represented as in ER diagram. Pre-condition rules check availability of external or transactional data and output of an activity. They are represented as conditions/constraints associated with an activity within curly brackets just below the activity name.

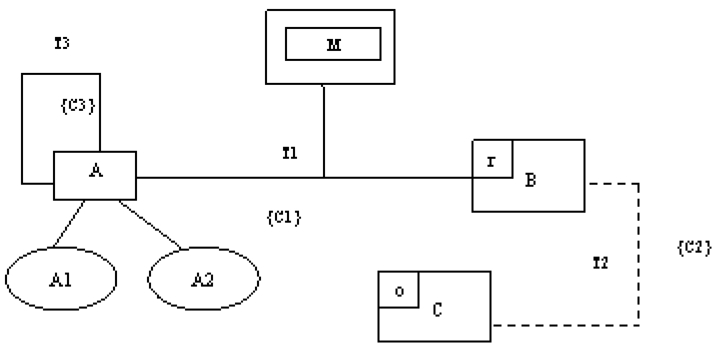


Fig. 6. Data Activity Model for Configuration

Figure 6 shows components of DAMC. *A* and *B* are two entities participating in activity *T1* and has pre-condition *C1* to be true. These entities represent master data. Entity *M* stores transactional data as a result of activity *T1*. *T2* is an optional activity and has pre-condition *C2* associated with it. For an optional activity *T2,* entity *B* is required entity whereas entity *C* is optional. Activity *T3* reads entity *A* under pre-condition *C3*. *A1* and *A2* are the attributes for entity *A*. *A* may have more attributes but only the relevant ones are shown in data model.

Configuration specific data requirements namely distinguishing between optional and required entities, distinguishing between persistent and transactional data, depicting optional and mandatory relationship and depicting business rules are met by DAMC.

5.2 An Illustration

DAMC corresponding to *create sales order* activity of EPC of figure 3 is shown in figure 8.

There are two types of Sales Order: *one time sales order* and *recurrent sales order.* A *sales order* is created for a particular *customer* involving some *material.* Both entities *customer* and *material* have persistent data whereas *sales order* entity has transactional data. *Sales order* is created if a *quotation is available* or if a *purchase order* is available. These are pre- conditions for *create sales order* activity to commence. Credit of a customer may be verified under any of the following conditions. For *one time sales order, credit* of the *customer* is verified whereas for *recurrent sales orders,* this is an optional activity. *Discount* may be associated with a *customer* depending upon *discount strategy or customer type.* For this optional relationship, entity *customer* is required whereas the entity *discount* is optional. *Tax* may be associated with *material* depending upon its *type* and/or *taxation strategy*. *Compute discount* and *Compute tax* relationship between entities *discount-pricing* and *tax-pricing* is optional and depends upon *discount status* and *tax status* respectively. The *pricing procedure* determines *pricing* of the material that uses entity *pricing* and *material. Pricing* uses entity *discount* and entity *tax.*

6 Validation of DAMC

Method Adoption Method (MAM) model [10] as shown in figure 7 is used to validate DAMC. The model evaluates a modeling technique on three parameters: perceived usefulness, perceived ease of use and intention to use. These parameters are explained next.

- Perceived Ease of Use (PEOU): It is defined as the degree to which a person believes that using a particular method would be free of effort.
- Perceived Usefulness (PU): The degree to which a person believes that a particular method will be effective in achieving its intended objectives.
- Intention to Use (ItU): It is the extent to which a person intends to use a particular method.

PU and PEOU are exogenous variables that have a positive impact on ItU. PEOU also impacts PU. We use survey method for evaluating our modeling technique. The questionnaire is adapted from Davis [3] to examine acceptance of the modeling methods. The aim of the questionnaire is to measure the three constructs namely PEOU, PU and ItU to find the acceptance of DAMC modeling methodology. There are six questions to measure construct PEOU, seven questions to measure construct PU and two questions measure construct ItU.

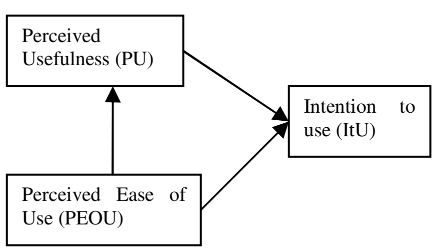


Fig. 7. Research Model

These items are based on measurement instrument proposed by Moody [10]. Items are formulated using a 5 point Likert scale, that varies from strongly disagree to strongly agree. Items were arranged in random order and half of the items were negated to reduce the monotonous responses to question items measuring the same construct. The following hypotheses are tested.

H1: *DAMC is perceived as easy to use for configuration.*
H2: *DAMC is perceived as useful for configuration*
H3: *There is intention to use DAMC for configuration*

To gather response to the questionnaire from industry personnel, multiple field visits were made. Respondents are mainly ERP consultants and program managers working for various ERP vendors like SAP, Oracle, MS Navision etc. Responses of 90 participants having ERP implementation experience in the range of 7-15 years are taken. Data analysis was done using SPSS 15.0.

An exploratory factor analysis using principal components factor analysis with varimax rotation was done to find the convergent validity of each predefined multi-item construct. Items that did not meet the cutoff value or were loaded in more than one factor were dropped from analysis. The loading of 0.50 or greater is regarded as significant [6]. For the construct PU, two items did not meet the criteria and were dropped from analysis. Cronbach alpha test is used to find overall reliability. We obtained the value of 0.9 and higher for all the three constructs. Alphas of 0.7 or above are considered to be acceptable in literature [12].

Table 1. One Sample t-test with significance level of 5% (alpha=0.05)

Statistic	PEOU	PU	ItU
Mean difference	1.037	1.028	1.063
95% conf. Interval for the diff	0.877 (lower)	0.895 (lower)	0.895 (lower)
	1.197 (upper)	1.156 (upper)	1.23 (upper)
T- value	12.89	15.443	12.636
1-tailed p-value	0.000	0.000	0.000

To verify H1, H2 and H3, we checked if the scores that the respondents assigned to the constructs were significantly better than the middle score on the Likert scale for an item. Scores of each subject was averaged over the different items that are relevant for a construct. One-tailed sample t-test was done to check the difference between the mean PEOU, PU, ItU and the middle score value 3. The results are shown in table 1.

The results strongly support rejection of null hypothesis. Thus, we empirically found that DAMC is perceived as easy to use and useful and that there is intention to use it in future.

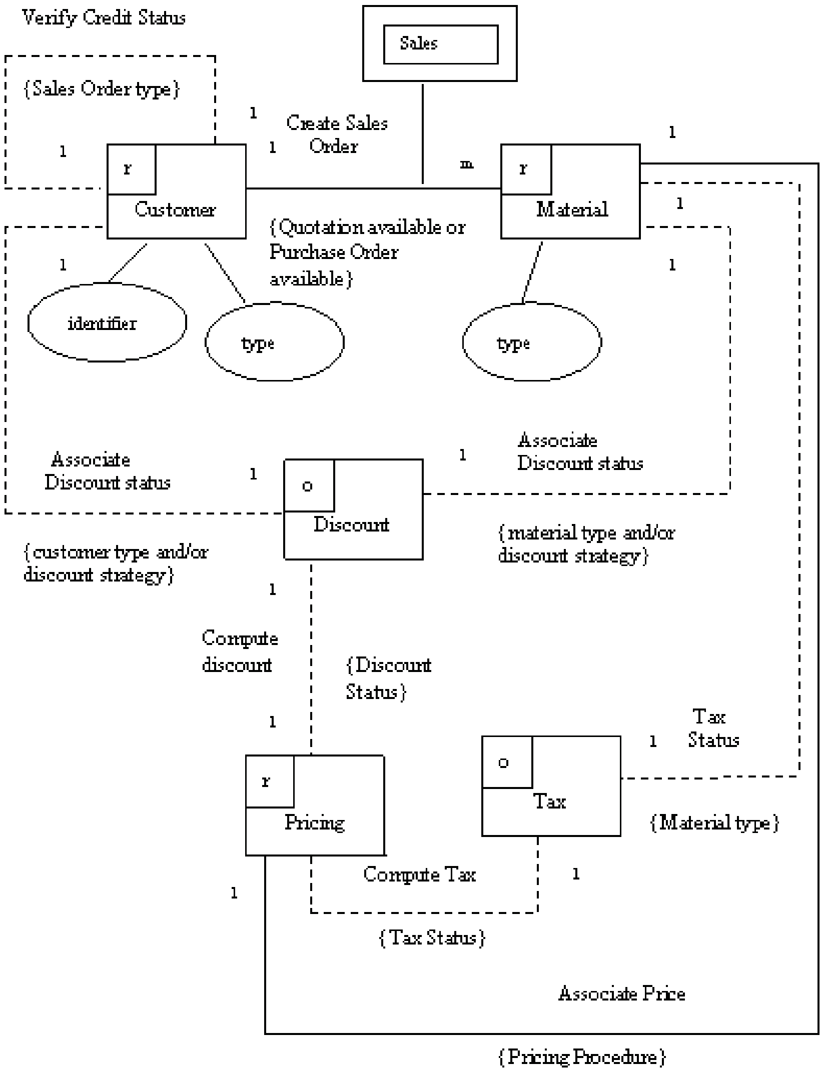


Fig. 8. DAMC corresponding to Create Sales Order activity

7 Conclusion

Configuration of ERP packages is a complex task. Many ERP implementations fail due to insufficient and incorrect configuration. One of the reasons for this failure is that configuration specific details are not captured while gathering requirements. A data modeling technique DAMC along with a process modeling method will facilitate configuration. DAMC is in its conceptual stages and is not used in industry. Therefore

ERP professionals have validated it to find the perception to use DAMC. The responses obtained are encouraging. Some of the respondents have expressed their desire to use this modeling technique for ERP implementation purpose. This will help us to find the actual usage of DAMC.

References

1. Bansal, V., Negi, T.: A Metric for ERP Complexity, Business Information Systems. Springer, Heidelberg (2008)
2. Booch, G., Rumbaugh, J., Jacobson, I.: The Unified Modeling Language Reference Manual. Addison Wesley, Reading (1999)
3. Davis, F.D.: Perceived Usefulness, Perceived Ease of Use and User Acceptance of Information Technology. MIS Quarterly 13(3) (1989)
4. Elmasri, R., Navathe, S.B.: Fundamentals of Database Systems, 3rd edn. Addison-Wesley, Reading (2000)
5. Eriksson, H.-E., Penker, M.: Business Modeling with UML: Business Patterns at Work. John Wiley & Sons, Inc., New York (2000)
6. Hair, J., Anderson, R., Tatham, R., Black, W.: Multivariate data analysis. Prentice-Hall, Upper Saddle River (1998)
7. Herbst, H., Knolmayer, G., Myrach, T., Schlesinger, M.: The Specification of Business Rules: A Comparison of Selected Methodologies. In: Verijn-Stuart, A.A., Olle, T.W. (eds.) Methods and Associated Tools for the Information System Life Cycle, pp. 29–46. Elsevier, Amsterdam (1994)
8. Luo, W., Strong, D.M.: A Framework for Evaluating ERP Implementation Choices. IEEE Transactions on Engineering Management 51(3), 322–333 (2004)
9. Markus, M.L., Arline, S., Petrie, D., Tanis, C.: Learning from adopters experiences with ERP: Problems encountered and success achieved. Journal of Information Technology 15(4), 245–265 (2000)
10. Moody, D.L.: The method evaluation model: a theoretical model for validating information systems design methods. In: Proceedings of the 11th European Conference on Information Systems, Naples, Italy, June 16-21 (2003)
11. Negi, T., Bansal, V.: Addressing the gap between requirements engineering and configuration in ERP systems. In: Proceedings of the 6th International Conference on Information Science, Technology and Management CISTM 2008, IIT Delhi, India, 31st July-2nd August (2008)
12. Nunally, J.: Psychometric Theory, 2nd edn. McGraw Hill, New York (1978)
13. Recker, J., Rosemann, M., van der Aalst, W., Jansen-Vullers, M., Dreiling, A.: Configurable Reference Modeling Languages. In: Fettke, P., Loos, P. (eds.) Reference Modeling for Business Systems Analysis, pp. 22–46. Idea Group, Hershey (2006)
14. Rosemann, M., Shanks, G.: Extension and Configuration of Reference Models for Enterprise Resource Planning Systems. In: Proceedings of the 12th Australasian Conference on Information Systems, vol. 1, pp. 537–546. Australian Computer Society Inc. (2001)

Designing and Developing Monitoring Agents for ERP Systems

Fabio Mulazzani[1], Barbara Russo[1], Giancarlo Succi[1], and Ernesto Damiani[2]

[1] University of Bolzano, Faculty of Computer Science,
Via della Mostra 4, 39100 Bolzano, Italy
{fabiomulazzani,barbararusso,giancarlosucci}@unibz.it
[2] University of Milan, Department of Computer Technology,
Via Bramante 65, 26013 Crema, Italy
damiani@dti.unimi.it

Abstract. This work discusses the case study of a bus company that needs to evolve its ERP system to enhance the managerial strategic control over the maintenance process of the bus fleet. This maintenance process was initially characterized by a poor performance that caused the low quality of the final service to the customer. The solution to the problem, we discuss in this article, is the integration of monitoring agents, called Probes, into the ERP system. Probes enable to control the process performance aligning it to the company's strategic goals. Probes offer a real time monitoring of the strategic goals achievement, also increasing the understanding of the maintenance activities. At the end, the use of Probes in the ERP system of the company has determined a significant change in the strategy of the company that obtained higher performances of the maintenance process and consequentially a higher quality of the offered service.

Keywords: ERP Systems, Monitoring Agents, Probes, Business Strategy, Business Processes.

1 Introduction

In Information Systems (IS), Enterprise Resource Planning (ERP) systems are a commodity in today's economy. ERP systems aim to integrate all the business processes into one enterprise wide solution [13, 14]. There is a large corpus of empirical evidence that reports of benefits by using ERP systems [5, 6, 11] or stories of failures [9, 10], indeed. Reasons of success are still under discussion, though.

In this paper, we present a case study of a successful enhancement of an ERP system of an Italian bus company. We pre and post - test for performance one of the critical Business Processes (BP) of the company before and after the use of monitoring agents we developed, called Probes.

Probes are in charge of automatically collecting appropriate metrics from the BP tasks, checking if the metric values are within a predefined range, and finally reporting back unusual behaviors to the strategic layer. We define them according to the

W. Abramowicz (Ed.): BIS 2009, LNBIP 21, pp. 240–251, 2009.

basic concepts of the Goal Question Metric (GQM) Plus Strategy method [1] and the Business Motivation Model (BMM) [12]. We then implement them as in the framework SAF [4]. With Probes managers can iteratively estimate the performance of the internal processes and performing corrective actions if needed. Employees can monitor their own work and the status of the business process, which they are working on, also by means of web-services [15], and tune their daily working pace if needed. Potentially, managers and employees can increase their knowledge and confidence in the ERP system.

This work represent the first application of SAF to a real case study that involved two years of data collection and analysis.

The rest of the paper is structured as follows. In section 2, we address the related works. In section 3, we offer an overview of the company used for our case study. In section 4, we describe the method used to conduct the case study. In section 5, we offer an analysis of the company performance before the introduction of the Probes included in the new functionalities. In section 6, we describe the evolution of the ERP system. In section 7, we analyze the company performance after the use of the new ERP system functionalities. In section 8, we conclude with some considerations and future works.

2 Related Works

A well-known method that aims to link business goal to process measurement is that of Balanced Scorecards (BSc) [8]. BSc were first developed with the purpose of measuring whether small scale operational activities of a company are aligned with its large scale objectives in terms of strategy. Nowadays, BSc are used a wider sense, which includes the alignment between strategic goals and business processes. Yet, they do not provide a constructive way to implement the strategy into the operational level. For example even though they use Key Performance Indicators (KPI) they do not discuss a method to derive and identify them.

Various factors during the deployment phase of an ERP system - such as top management commitment, reengineering, integration, implementation time, and implementation costs [2, 3] - may cause a low achievement of the benefits promised by the adoption of such a technology. Among these factors it has to be considered also the lach of the ERP System in monitoring the process performance and others critical success factors. Many of these causes can be determined by the fact that ERP systems are not easily customizable. Namely, ERP systems are built on the best practices followed in the industry. These are based on industry knowledge gained from several deployments all across the world [7]. As a consequence little space is given to specificities. Namely, sometimes, business processes and the relative strategies are so unique that any standard does not represent them correctly. Research studies have shown that even the best application package can meet only 70% of the organizational needs [16]. To increase this percentage ERP vendors usually suggest to adopt a prescribed model to implement their ERP products [7, 17], thus, fitting business processes into the ERP solution more than adopting the ERP technology to represent the organizational business settings.

3 The Case Study

Dolomitesbus is (a pseudonym for) a public transportation company operating in a province of Northern Italy. Dolomitesbus has a bus fleet of 290 units, and serves 60 routes connection every day. Each route is covered by two buses, one that works from 6 a.m. to 2 p.m., the other from 2 p.m. to 10 p.m.

The mission of the company is to offer to the customer a high quality transportation service over the province territory. The fleet is subjected to an extreme mechanical wear due to the mountain routes that serves, hence the main issue for the company is to concentrate a lot of resources on the maintenance process in order to have efficient buses. The mechanical workshop, that is part of Dolomitesbus, is in charge of any type of maintenance operation (i.e. planned maintenance or damage repair), and it also has to guarantee at least two fully operative buses every day for each route in accordance to the main quality constraint given by the Provincial Council - that is the supervisor for the company activities. The maintenance tactic imposed by the managers is the following: “Make a planned maintenance to every bus at most once per year - hence respecting the minimum requirement of the law - every other maintenance has to be made only when a problem occurs and needs to be repaired”.

Furthermore the company has defined a feedback mechanism where the customers can report their complaints. Till the end of 2007 the company has never inspected and properly used the resulting complaints.

For the last decade, the buses capacity reaches its critical load in the period of June / August. As such, the high number of passengers that are served in this period affect the company’s activities in two ways: (i) increase the mechanical waste of the buses - due to the increase of the weight loaded; (ii) increase the attention over the quality of the service offered - due to the increase of the customers transported.

Dolomitesbus has an efficient IT department that has always supported all the company’s software needs also adopting and developing Open Source Software. At the time the case study started, the company had its own in house ERP system that covered the activities of its departments but the maintenance process in the workshop. The operations of maintenance and repair done over a bus where only recorded manually with a paper schedule (containing fields such: bus number, type of operation, operation starting date, operation end date, details, etc.), not always properly filled, and then stored into an archive.

4 Experimental Protocol

The research was conducted from October 2007 to August 2008 and divided into three phases.

During phase one, that was carried out in October 2007, (i) we studied and analyzed the company context by means of field interviews to junior and senior CIO/CEO managers, and to the workshop manager; (ii) we prepared the GQM to understand and characterize the current maintenance; (iii) we converted to a digital format the data contained into the maintenance paper schedule, then we analyzed the maintenance process performance metrics - for the years 2006 and 2007 - in accordance to the GQM defined in the previous point; (iv) based on the analysis of the data collected, we reported our considerations to the company managers.

During phase two, that was carried out from November to December 2007, (i) we developed the Probes for the company according to the new BMM defined by the managers after the results of phase 1; (ii) based on the new BMM, we suggested Dolomitesbus IT Department to develop new ERP functionalities that included the Probes and we then follow their development. The new functionalities introduced into the ERP system enabled to: (i) Fill the blank fields previously required by the paper schedule into a web module so that all the information where stored into the common ERP database, and furthermore giving the constraint that all the fields required were properly filled; (ii) Record in the ERP Database the customers complaints, so that it could be adopted as one of the indicators for the quality of the maintenance process; (iii) Integration of the Probes to support the maintenance of the process to support the maintenance process and to control the achievement of the strategic objectives.

During phase three, that was carried out from January to September 2008, we measured and analyzed the maintenance process performances after the introduction of the new ERP system functionalities, successfully deployed by the IT Department.

In the following Sections we illustrate the action taken and the results obtained during the three phases.

5 Phase 1 – Measuring the Initial Degree of Performance

As the first step of this phase we interviewed the main responsibles of Dolomitesbus in order to better understand the various aspects of the company context in the maintenance process. For this we developed the GQM to characterize the maintenance business process. The Measurement Goal of the GQM are the following:

(1) Measurement Goal: Analyze the bus maintenance process in the context of the company workshop to characterize it in terms of number of buses under maintenance;

(2) Measurement Goal: Analyze the bus maintenance process in the context of the company workshop to characterize it in terms of time spent to operate over a bus.

The information needed to answer to the questions of the GQM were retrieved from the paper schedule. As mentioned, the schedule was designed with the following fields to be filled:(i) Bus number; (ii) Current Km; (iii) Operation Starting/Ending Date (hh.dd.mm.yyy); (iv) Total hours spent for the operation; (v) Type of operation - checkbox chosen between planned maintenance and damage repair; (vi) Detailed description of the operations; (vii) Mechanic responsible for the operation.

The paper schedule was used with the purpose of having an historical record of the operations done over the buses.

During the conversion to the electronic format of the paper schedules we found out that some fields were not properly filled or not even considered at all. The fields left blank were: Current Km; Hours of starting and ending date; Total hours spent for the operation. The fields not properly filled were: Detailed description of the operations - when filled it included a list of material used for the operation instead of a detailed description of the problem encountered; Mechanic responsible for the operation - it included an unreadable signature but not the cursive name and surname. The fact that the workshop mechanics have never filled the schedules in a proper way prove the fact that these have received a scarce importance or consideration. Namely, the schedules have never been used for any managerial analysis to establish the performance of the maintenance process. Furthermore, the bad compilation of the schedules has determined the permanent loss of important information that would have been useful for our analysis.

The results obtained by the application of the GQM are summarized in Table 1.

Table 1. Metrics of the GQM collected

Description (PM: Planned Maintenance; . MfD: Maintenance for Damages)	2006	2007	'06/'07
1.1 Question: How many maintenance operations are done each day divided by type of operations?			
Mean number of buses daily operated for PM	40	33	41
Median number of buses daily operated for PM	45	48	40
Mean number of buses daily operated for a MfD	90	91	101
Median number of buses daily operated for a MfD	115	111	100
1.2 Question: How many maintenance operations are done each year divided by type of operation?			
Total number of buses yearly operated for a PM	370	330	350
Total number of buses yearly operated for MfD	2360	2200	2280
2.1 Question: How much does each type of operation last?			
Mean duration in days for operations of type PM	39	69	52
Median duration in days for operations of type PM	7	16	9
Mean duration in days for operations of type MfD	13	20	16
Median duration in days for operations of type MfD	1	1	1

We then reported to the company managers our suggestions, as reported in detail in the following section.

5.1 Data Analysis

From the data collected we developed and reported to the managers our first results and suggestions.

The data collected for Metrics 1.1.1 shows that the median number of buses daily operated is 40 for those under a planned maintenance and 100 for those under a maintenance for damages. In figure 1 is represented a line chart of (i) the daily number of buses having planned maintenance, (ii) the daily number of buses having a maintenance for damages, and (iii) the sum of the previous two. In figure 1 we also show the threshold of the maximum number of buses that should be under maintenance every day without affecting the company mission. Dolomitesbus should have 120 buses (2 buses for each of the 60 routes) fully maintained every day in order to offer to its customers a high quality service, hence the maximum number of buses under maintenance every day is 170 (given by the 290 buses of the fleet minus the 120 fully maintained each day).

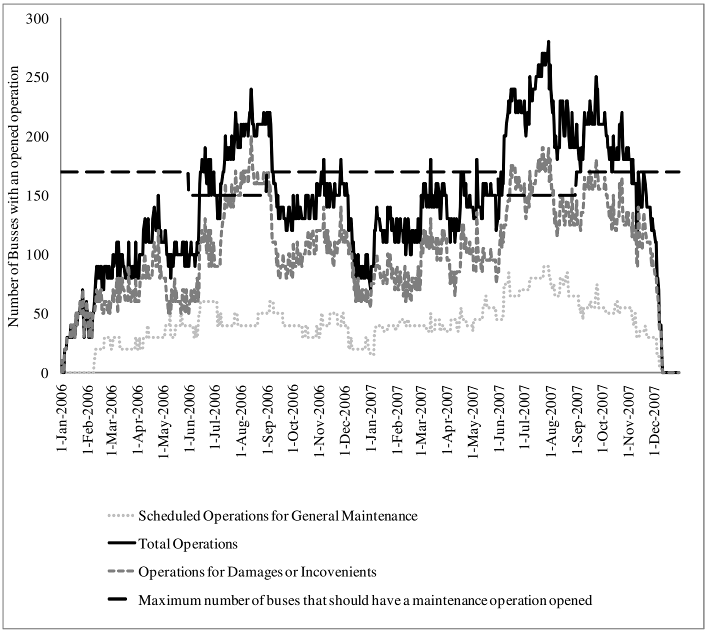


Fig. 1. Number of Maintenance Operations done in 2006 and 2007

From June until the end of August the threshold is of 150 buses, since the high number of customers in that period requires an increase to 140 buses fully maintained each day. If we consider these two levels for the thresholds as a quality constraint over which the quality of the service is not guaranteed, we counted that in 2006 the limit was exceeded for 63 times (55 during summer time and 8 during the rest of the year), while in 2007 the limit was exceeded for 151 times (85 during summer time and 66 during the rest of the year). That means that for 214 days during the biennium 2006-2007 the routes were served with buses not completely maintained and that could have negatively affected the customers' quality perception. As confirmed by the managers feedbacks, during that particular days some of the 60 routes where covered by buses with a maintenance operation opened.

The data collected for Metrics 1.1.2 shows that in the years 2006 and 2007 the mean of total planned maintenance operations is 350, and the mean of total operation for maintenance for damages is 2280. Considering these numbers over the bus fleet we can notice that every year each bus is subjected to a mean of 1,2 operations for planned maintenance and 7,6 operations for maintenance for damages.

The data collected for Metrics 2.1.1 shows that the mean duration for the planned maintenance operation is 52 days, while the mean duration for the operation of type maintenance for damages is 16 day. That means that every bus spent a mean of 69 days per year with an unsolved inefficiency - calculated multiplying the mean duration of the operation per the mean number of operation that a bus is subjected each year. Figure 1 clearly shows a different trend on the number of maintenance during 2006 and 2007. The managers of Dolomitesbus expected a difference between the two years, and they knew the causes. In fact, during an inspection at the company workshop at the end of 2006 they found out that the workshop managers did not spend much attention in filling the maintenance paper schedule. After the inspection, the top management imposed to the workshop manager more accuracy in filling the schedule and wrote down and defined a first attempt of business strategy related to the maintenance process. In section 6.1 in figure 2 we modeled the business strategy for the year 2007.

6 Phase 2 – The ERP Probes

In this phase we offer to Dolomitesbus some suggestions on how to enrich the existing ERP system with (i) new functionalities oriented to support the bus maintenance process, and above all, (ii) with the Probes that monitor the related manager's strategies.

When Dolomitesbus IT Department developed the new functionalities mentioned above included into them the Probes that helped the managers to monitor the achievement/respect of the strategic goals or constraints that they defined.

The development of the Probes for the process metrics collection was done according to the principles of the GQM^{+}Strategy. In section 6.1 we illustrate the BMM and the Probe development in our case study.

Probes are agents that aim to increase the company performances through the enhancement of the strategic goals monitoring. The Probes are integrated into the ERP system and are in charge of the following actions:

1. Collecting specific metrics generated by the tasks of the business process;
2. Checking if the metrics are within a predefined range;
3. Reporting the metrics analyzed and highlight if their value is outside the predefined range.

Into an ERP system the collection of specific data or metrics (action 1) can be done either by querying the ERP common database and or reading the log file produced by the tasks of the workflows. The database can be queried to extract the data filled by an operator of the ERP or elaborated by the ERP, while the log files is read to extract information like the time of activation, the end of use of a task or others.

The second action is to verify if the metrics are within a predefined range that is determined from an analysis of the company business strategy. The strategy threshold can be given in the following ways:

1. Statically by the manager - In this case the manager responsible for the process monitored by the Probe indicates a threshold according to its managerial experience.
2. Historical trend - In this case the threshold is determined according to the historical trend of the process. That requires that the process has been monitored in the past with appropriate metrics.
3. Mathematical reference models - Derived from the literature on the sector.

The last action performed by the Probes is to report the metrics analyzed offering to the managers a view of the achievement of their desired result or the respect of the given constraints. In this action the Probes offer a value added to the managers, compared to the usual KPIs, managing to link the performance metrics to the strategic goals or constraints.

6.1 The Probes Development

After the managers defined a new business strategy for the maintenance process and after Dolomitesbus IT Department implemented the new functionalities into the ERP system we defined the type of Probes useful for the managerial control.

To develop the Probe we first identified the business strategy that governs the maintenance process by modeling it with a standard notation called Business Motivation Model as described in Fig. 2. On the top of the figure is represented the mission of the company that can be achieved by means of some strategies. In the figure we included the specific strategy that refers to the maintenance process. On the bottom of the figure we then described the tactics, and the relative constraints, that the company needs to respect to fulfill the above strategy. On the left side we show the tactics used during the year 2007, while on the right side we show the tactic developed after our intervention at the end of 2007.

For the year 2008 the managers decided to adopt three new tactics with the relative constraints related to the maintenance process. Here follows the list of the new tactics and constraints as reported in figure 2:

- Tactic one (T1) imposes that of the 290 buses of the fleet only 240 are maintained by Dolomitesbus workshop, the remaining 50 needs to be maintained by a different workshop. The constraint for this tactic tells that the external workshop must guarantee at least 40 buses fully maintained every day, while the internal workshop must guarantee at least 90 buses fully maintained every day.
- Tactic two (T2) imposes that in case one of the threshold given in the constraint for the tactic one is not respected then two or three routes must be aggregated in couples or triples. The aggregated routes must be served by a bus that works over its time limit.
- Tactic three (T3) imposes that the manager must revise tactic one in case the company receives too many customers complaints. The constraint for this tactic is given by a threshold risk of 5%.

We have adopted as additional overlayer of the GQM^+Strategy the tactics that the managers defined for 2008 and we have then applied the GQM method to determine the metrics to be monitored. The logic of the threshold for the reporting function of the Probes has been derived from the business constraints that governs the relating tactics. Here it follows the structure of the GQM that determines the metrics to be measured by the Probes and the related threshold.

-Probe 1 - Measurement Goal (MG) for T1: Analyze the maintenance process from the point of view of the managers in order to understand the level of maintained buses in the context of Dolomitesbus Workshop.

Question 1.1: How many buses are under maintenance at Dolomitesbus workshop every day divided by typology of maintenance?

Metric 1.1.1:(Absolute) Number of Buses by day and typology.

Threshold for this metric: According to CfT1 the max limit is 150.

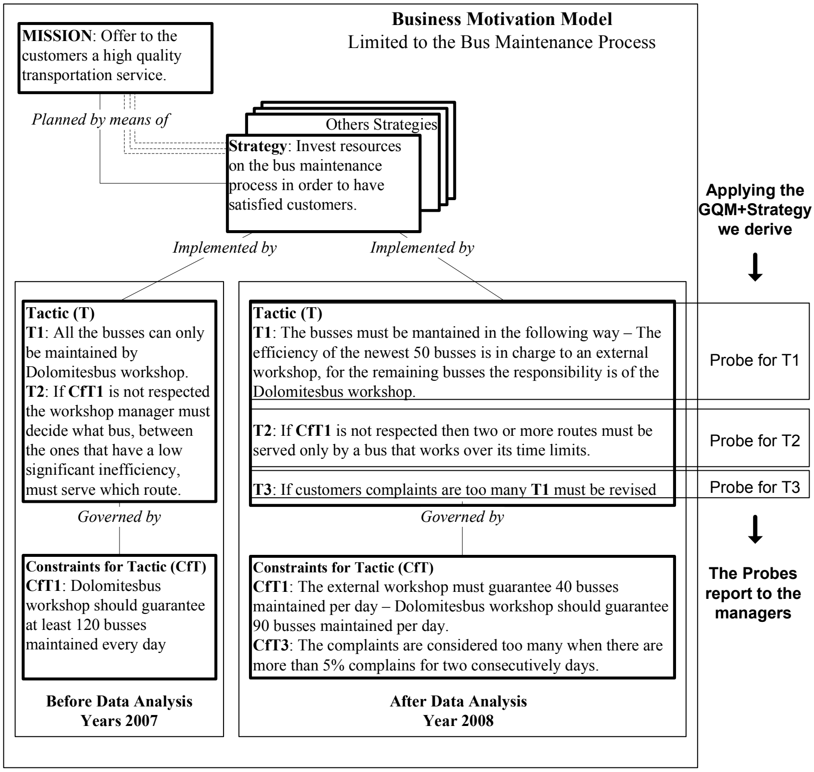


Fig. 2. Defining Probes

-Probe 2 - MG for T1: Analyze the maintenance process from the point of view of the managers in order to understand the level of maintained buses in the context of External Workshop.

Question 2.1: How many buses are under maintenance at the external workshop every day divided by typology of maintenance?

Metric 2.1.1: (Absolute) Number of Buses by day and typology.

Threshold for this metric: According to CfT1 the max limit is 10.

-Probe 3 - MG 1 for T2: Analyze the maintenance process from the point of view of the managers in order to understand the level of aggregation of bus routes caused by the unavailability of maintained buses (if there are not enough buses available the routes can be aggregated in couples or triples and served by a bus that works over it time limit).

Question 3.1: How many bus routes are aggregated each day distributed by couples or triples?

Metric 3.1.1: (Absolute) Number of bus routes

Threshold for this metric: The max limit can be defined after a simulation.

-Probe 4 - MG 1 for T3: Analyze the maintenance process from the point of view of the managers in order to understand the level of customers complaints.

Question 4.1: How many complaints there are every day?

Metric 4.1.1: (Absolute) Number of Complaints

Threshold for this metric: Below a given threshold risk (i.e. 5%).

Probes were developed using the model in figure 2. It is now under development a functionality that will send a warning mail to the managers if the indicators reach a below-threshold value.

7 Phase 3 – The New Business and IT Set Up

In this phase we measured the performance of the maintenance process during the first three quarters of 2008.

Table 2. Metrics collected by the Probes (Sec. 6.1) in the first three quarters of 2008

Probe	Metric	PM '08	MfD '08	PM+MfD '08	'06/'07	Δ
Probe 1	Metric 1.1.1	20	50	70	N.A.	N.A.
Probe 2	Metric 1.1.2	6	4	10	N.A.	N.A.
P. 1 & 2	Metric 1.1.1+1.1.2	26	54	80	140	-43%

Table 2 shows the metrics values collected by the probes developed in section 6.1. The values collected in 2008 were compared, if possible, to the average value of 2006/2007.

Probe 1 has measured that every day inside Dolomitesbus workshop there is an median of 70 buses that are subjected to a maintenance process.

Probe 2 has measured that every day inside the external workshop there is a median of 10 buses that are subjected to a maintenance process.

Summing the metric collected by the Probes 1 and 2 we observe that the adoption of new business strategies defined by the company managers have determined a drastic decrease of 43% of the buses that are under any type of maintenance every day. That means that the choice of externalizing part of the maintenance process has determined a decrease not linear of buses present in the workshop every day for a maintenance. Furthermore, during 2008 Dolomitesbus have never exceeded the maximum threshold of buses that daily could be under maintenance.

Probe 3, that is not represented in table 2, reported that no routes were aggregated neither in couples nor in triples. That is due to the fact that Dolomitesbus managed to constantly have all the efficient buses required. This metric is derived from the new strategy adopted for 2008, so it is not possible to compare this value to any previous year.

Probe 4, that is not represented in table 2, reported that no complaints were presented from the customers. The quantity of complaints collected by this Probe during 2008 cannot be compared to those of 2006/2007 since in these years the company has never stored the complaints received.

The performances measured are the results of the combination of the following factors:

1. The analysis of the maintenance process taken in phase 1;
2. The adoption of significant changes (i) on the company business strategy concerning the main-tenance process, and (ii) on the relating business process;
3. The development of new functionalities having the peculiar characteristic that included the Probes for the existing ERP system.

8 Conclusion and Future Works

In this paper we present a case study of a transportation company that has evolved its ERP system with new functionalities. The new functionalities has been developed to support a critical issue such the bus maintenance process.

The peculiarity of this case is the adoption of monitoring agents called Probes integrated into the new functionalities. The Probes are agents that monitor the process performances and reports to the managers a quantitative measure with respect to the strategic goals, objectives or constraints. The Probes has been developed using the GQM$^+$Strategy software engineering technique principles.

We demonstrated that the Probes are one of those factors that could increase the benefits obtained by using ERP systems with an automated action of the managerial control responsibilities.

It remains clear that the benefits obtained by the Probes can be limited if not combined with a manager intervention. In fact, we noticed that the better performances of the critical business process reached in 2008 by the company of our case study have been obtained partially because of the new ERP system functionalities implementation including the Probes, and partially because managers adopted a new and appropriate business strategy.

So far, we determined how to develop the Probes so that the business process performances could be linked to the business strategy. The main issue on deploying the Probes is that the system where they operate needs an upfront customization. As such, one of the future topic of research will be the automation performed by defining the business strategy with a controlled natural language, and checking and reporting the strategic goals, objective and constrains through an ontology reasoner.

References

1. Basili, V., Heindrich, J., Lindvall, M., Münch, J., Regardie, M., Rombach, D., et al.: Bridging the Gap Between Business Strategy and Software Development. In: International Conference on Information Systems Management, Québec, Canada (2007)
2. Bingi, P., Sharma, M.K., Godla, J.K.: Critical Issues Affecting an ERP Implementation. Information System Management, 7–14 (1999)
3. Ciolkowski, M., Heidrich, J., Muench, J., Simon, F., Radicke, M.: Evaluating Software Project Control Centers in Industrial Environments. In: First International Symposium on Empirical Software Engineering and Measurement, pp. 314–323 (2007)

4. Damiani, E., Mulazzani, F., Russo, B., Succi, G.: SAF: Strategic Alignment Framework for Monitoring Organizations. In: Abramowicz, W., Fensel, D. (eds.) BIS 2008. LNBIP, vol. 7, pp. 213–226. Springer, Heidelberg (2008)
5. Davenport, T.: Mission Critical: Realizing the Promise of Enterprise Systems. Harvard Businss School Press, Boston (2000)
6. Davenport, T., Harris, J., Cantrell, S.: The Return of Enterprise Solutions: The Director's Cut. Accenture (2002)
7. Kale, V.: Implementing SAP R/3: The Guide for Business and Technology Managers. Sams (2000)
8. Kaplan, R.S., Norton, D.P.: The Balanced Scorecard: Translating Strategy into Action. Harvard Business School Press, Boston (1996)
9. Kumar, V., Maheshwari, B., Kumar, U.: An investigation of critical management inssues in ERP implementation: empirical evidence from canadian organizations. Technovation, 793–807 (2003)
10. Markus, M.L., Benjamin, R.I.: Change agentry - the next IS frontier. MIS Q, 385–407 (2005)
11. O'Leary, D.E.: Enterprise Resource Planning Systems: Systems, Life Cycle, Electronic Commerce and Risk. Cambridge University Press, Cambridge (2007)
12. OMG. Business Motivation Model (BMM) (2006), http://www.omg.org/docs/dtc/06-08-03.pdf
13. Pearlson, K.E.: Managing and Using Information Systems. Wiley, New York (2006)
14. Powell, C., Dent-Michallef, A.: Information technology resources. Strategic Management Journal, 375–405 (1997)
15. Predonzani, P., Sillitti, A., Vernazza, T.: Components and Data-Flow Applied to the Integration of Web Services (IECON 2001). In: The 27th Annual Conference of the IEEE Industrial Electronics Society, pp. 2204–2207. Denver, CO (2001)
16. SAP AG. Integrated Industry-Specific Enterprise Solutions for Midsize Businesses. Brochure Material (2005)
17. Wells, A.J.: Oracle 11i E-Business Suite from the Front Lines. CRC Press, Boca Raton (2003)

Organizational Implications of Implementing Service Oriented ERP Systems: An Analysis Based on New Institutional Economics

Jan vom Brocke, Bernd Schenk, and Christian Sonnenberg

Hilti Chair of Information Systems and
Business Process Management,
University of Liechtenstein, Fuerst-Franz-Josef-Strasse 21,
9490 Vaduz, Principality of Liechtenstein
{jan.vom.brocke,bernd.schenk,
christian.sonnenberg}@hochschule.li

Abstract. The promise of Service-oriented Enterprise Resource Planning Systems is to allow for a fine-grained alignment of IT with the specific functional requirements of an organization's business processes. Recent publications suggest that the implementation and configuration of service-oriented ERP systems are predominantly coordinated by means of market-based transactions, since services are supposed to be offered via service marketplaces. However, emphasizing the significance of business processes as a starting point for any service-oriented information system implementation, we suggest that the implementation process surprisingly demands an even higher degree of hierarchical coordination compared to conventional ERP system implementations.

Keywords: Service oriented architectures (SOA), Enterprise Resource Planning (ERP), Implementation Process, New Institutional Economics, Transaction Costs Theory, Principal-Agent Theory.

1 Introduction

Service oriented architectures (SOA) herald a new generation of application systems. Whereas current systems were to be chosen and operated as self-contained systems of individual vendors, service oriented architectures allow for integrating application functionality from different sources internal or external to a company according to individual needs within a specific context. The basic idea of SOA – which this paper is focusing on – is the increased possibility of IT-Business Alignment that is enabled by loosely coupled systems. Service orientation enables the realization of "best-of-breed" ERP systems [1] where Web service based software modules of different vendors are connected to a system that supports the business processes of an applying organization.

As the number of service oriented ERP systems increases, SOA becomes a relevant and important subject for companies to decide upon. GARTNER predicts a massive increase in Web service enabled IT-services in the upcoming years [2]. Such

W. Abramowicz (Ed.): BIS 2009, LNBIP 21, pp. 252–263, 2009.

IT-services will – according to a Gartner survey – reach a market volume of 261 billion US-Dollars at the beginning of the year 2009. This represents 30 % of the worldwide IT-service market.

When managing the inherent complexity of a service oriented ERP system properly, the expected benefit is estimated to be significantly higher when compared to conventional ERP systems [3]. The deployment of service oriented architectures also changes the role of the partners in the ERP implementation process [4]. The well-established and structured process chain between software vendor, consulting company and applying organization known from conventional implementation projects must be reconsidered in its allocation of tasks. The division of conventional, so-called "monolithic" [3] systems into independent and loosely coupled available services (i.e. application components intended to be exchanged on a market) changes the implementation tasks.

New versions of existing ERP suites are technologically based on SOA. However, this fact does not qualify them to be a service oriented ERP system in the sense of this paper. These web-service enabled suites can be seen as a *hybrid SOA* solution that enables applying organizations to integrate existing third-party applications. However, they remain ERP suites that work best when only components of the same vendor are deployed because they contain proprietary elements. These ERP systems do not allow for the integration of non-vendor specific services, resulting in a tight coupling with the initial vendor.

This is an important aspect to keep in mind when dealing with the implementation of service oriented ERP systems. In the case of hybrid-SOA based ERP suites the implementation process does not differ significantly from the implementation process of conventional ERP systems. As for our analysis, these suites are therefore considered conventional ERP suites.

A *fully service oriented* ERP system allows for the integration of services from any kind of service provider. With fully service oriented ERP systems the exchange and integration of know-how – which is considered as the main implementation transaction – is split up into many smaller transactions (see Fig. 1). For selecting and

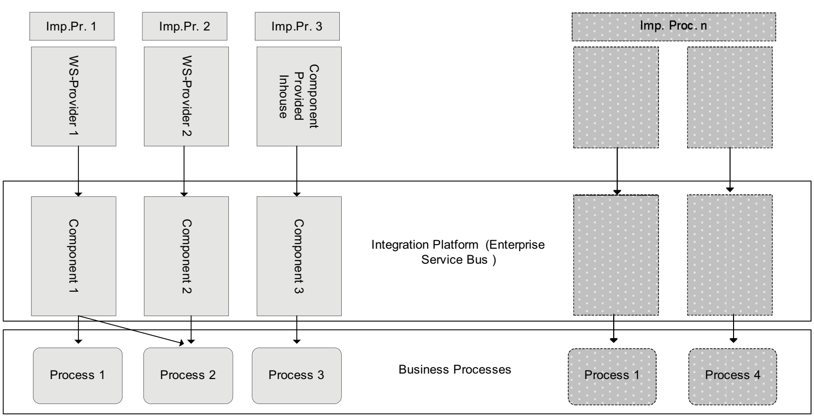


Fig. 1. Business Process-driven Implementation Process

integrating particular services both technical as well as business process-oriented know-how is required. These know-how requirements are expected to differ among the various implementation processes since different business needs, business processes and service provider have to be considered.

As the implementation process of conventional ERP systems is based on the ERP-software and its built-in pre-defined business processes, we therefore name the implementation "software-driven" for the purpose of our analysis. In contrast, the implementation process of service oriented ERP systems overcomes this software-driven approach. The nature of service oriented systems lies in their orientation to business processes. This enables the design of an ERP system which is much more closely aligned with the business process requirements in a specific situation. As stated above, service oriented ERP systems enable a "best-of-the-breed" approach that allows the combination of software modules from different software vendors according to varying business process requirements. This should result in an increasing flexibility and adaptability of service oriented solutions compared to conventional ERP systems.

In the implementation process of a service oriented ERP system, the first step must be the analysis of business processes in the applying organization. In a second step the market for Web services is analyzed and suitable software services for single steps of a process are chosen [5]. Therefore every module (i.e. service) can be evaluated according to its suitability to the individual task. By having the options both to choose among several service alternatives as well as to integrate service at run-time, service oriented ERP systems as such may contribute to an increase in value for the applying organization.

However, instead of analysing the economic consequences of operating service oriented ERP systems (cf. [6]) for an evaluation methodology) the aim of this paper is to identify and characterize organizational implications of the service oriented paradigm on the implementation process. Taking the perspective of the New Institutional Economics [7] suitable forms of interaction between consulting company and applying organization in the implementation process of ERP systems are analyzed.

As a research method we mainly rely on a literature review complemented by conceptual-analytical reasoning. As we are in an early stage of our research we consider this to be a suitable practise to start with. To our knowledge, little research has been conducted on this topic so far. We therefore analyse existing literature in the field of software implementation of service oriented ERP systems and synthesize these findings by applying the perspective of New Institutional Economics.

The paper is structured as follows. First, an analysis of related work on service oriented ERP systems is provided. Subsequently, the fundamentals of New Institutional Economics are presented. Based on this, different types of transactions between the partners involved in the implementation process are analyzed. The findings of this analysis are synthesized by means of qualitative assessments and possible modes of organizing the implementation process of service oriented ERP systems are presented. The paper concludes with summarizing the major results and highlighting the demand for further research.

2 Related Work and Theoretical Foundation

Within the literature many contributions on the topic of implementing ERP systems can be found. Besides proposing critical success factors [8], [9] much attention is being paid to the integration of end users and increased acceptance of the implemented solutions [10], [11], [12]. The adaptation of the ERP system by means of customizing parameters and extended adaptation like the development of add-on programs at defined interfaces in the software is frequently addresses within the literature [13], [14], [15]. Furthermore the realization of competitive advantages through a *business process-oriented* implementation of ERP systems is analyzed [16], [13].

When implementing service oriented ERP systems organizations are not limited to adapting their existing processes to pre-defined process blueprints that are inherent to the ERP solution (i.e. conventional ERP systems). Instead they flexibly integrate suitable software services in alignment with existing business process requirements. Customization of ERP systems is now conducted by means of business process engineering tasks, allowing for the configuration of organizational rather then technical parameters [5]. However, the principle of loose coupling of a SOA increases not only the flexibility but also the complexity of such a solution. The complexity rises with the increasing number of modules or software services that are to be integrated. In particular, complexity increases due to increased integration efforts, error handling and business process compliance concerns. The question is if the rising complexity of deploying a service oriented ERP system is outweighed by the benefits of the SOA solution, therefore justifying an investment in a SOA initiative [17], [3], [18], [6].

Furthermore, as SOA opens up the possibility of integrating third-party software vendors, leading software companies try to protect their dominant position in the market by developing enterprise service buses which contain proprietary components (concerning the case of SAP cf. [19], [20]).

As a SOA implementation turns out to be a complex task, suitable forms of coordination of the implementation process have to be selected. As stated above SOA enables manifold ways of exchanging implementation services which also encompasses market-based transactions. However, the impact of service oriented architectures on the selection of suitable coordination mechanisms for the implementation process has hardly been considered as yet. This is amazing as both researchers as well as practitioners underpin the paramount importance of SOA in the ERP area [21], [22], [19].

New Institutional Economics [23], [10], [7] serves as a means of identifying suitable forms of organizational arrangements for the execution of transactions and the exchange of property rights. In particular, it explains economic relationships in a world of imperfect actors characterized by bounded rationality and opportunism [24].

As our aim in this paper is to examine the differences between conventional implementation processes and implementation processes within a service oriented context we will first use transaction cost theory to analyze alternative forms of interaction between stakeholders of the implementation process and to evaluate the efficiency of these interaction forms [25], [19]. As a second main theory of New Institutional Economics the Principal-Agent Theory (PAT) is chosen to analyze the implications of asymmetric information distribution among the stakeholders throughout a service oriented ERP system implementation. As described above, the change from a standardized software-driven to a very specific business process-driven implementation process changes the specificity of the information to be exchanged. PAT contributes to understand the economic implications of information problems due to information asymmetry as well as specific paradigms and strategies applied to overcome these problems [26].

A transaction covers all activities that must be performed to enable the exchange of goods and services [24]. For the purpose of this paper we will focus on the exchange of implementation services. Activities that are necessary to produce these services are therefore not part of a transaction. For the execution of transactions two dichotomous forms of organization are considered: market and organization [23]. When applying a market-based exchange individual agreements for every exchange of services have to be made. The price of the service acts as the coordinating mechanism. Within an organization long-term agreements substitute the individual agreement for every exchange. The coordinating mechanism in this case is the directive provided by the organizational hierarchy. Market and organization mark opposing end points allowing for hybrid organizational forms of service exchanges in between.

PAT analyzes the exchange of services from the perspective of contractual agreements [26]. The roles within this exchange are the agent and the principal. Usually the agent is better informed than the principal resulting in a risk of a welfare loss for the agent.

If information is available at no cost, optimal or so-called First-Best solutions can be achieved (i.e. the service exchange guarded by the contractual agreements lead to an increase in welfare for both parties). However, in reality some hindering influence factors can be observed. Limited availability of information and opportunistic behaviour of the contracting parties lead to suboptimal solutions – so called Second-Best solutions. These solutions are characterized by a high cost to gain information and a loss of welfare for the principle due to information asymmetries. In real life conditions agents can behave opportunistically as their behaviour cannot be completely monitored.

The term agency cost describes the difference between the value of possible First-Best and Second-Best solutions. According to the three information problems considered within the PAT (i.e. Adverse Selection, Moral Hazard and Hold-Up), three types of agency costs can be identified [26]:

- Signalling cost of the agent (Adverse Selection).
- Monitoring cost of the principal. (Moral Hazard):
- Remaining loss in welfare (Hold-Up).

When considering the Principal-Agent Theory for the design of organizational arrangements, a major design objective should be to minimize the agency cost by finding the Second-Best solution that comes closest to the First-Best solution.

The contribution of the PAT is the integrative analysis of questions concerning the division of labour and specialization on the one hand and questions of exchange and agreement on the other hand. It explicitly takes into account the cost of suboptimal solutions. Not-concluded agreements because of prohibitive cost of information are also covered. Furthermore, PAT provides a link to transaction costs, as the costs of initiating a contract (i.e. transaction costs) are covered by signalling and monitoring costs. Through the application of PAT, suitable organizational settings can be identified in order to reduce unexpected actions of contracting parties and therefore to minimize agency cost.

3 Implications of New Institutional Economics

As our analysis is based on new institutional economics theory it is of great importance to clearly define the transaction that we focus on in this paper. Considering the implementation process of ERP systems the transaction is then defined as:

The transaction of property rights on know-how that is useful for getting an existing application running in a specific applying organization.

The know-how that is subject to this transaction can be divided into skills and methods - where skills are targeting human resources with their inherent implementation knowledge and methods are seen as well described procedures that have been proven in prior implementation processes. Know-how is in this case not only technical knowledge about the application and its integration but furthermore knowledge about the specific industry and its special processes and needs. These specificities also have an influence on the design of the implementation process.

For evaluating different forms of organizing the implementation process, the characteristics of transactions as stated in the Transaction Cost Theory have to be considered:

- *Characteristic of Specificity*: Investments in human capital are required to build up the necessary knowledge of methods and domains [7]. The implementation of service oriented ERP systems requires a higher degree of specific investment in know-how for business process analysis. Prior to the technical implementation, business processes have to be analyzed and selected according to their specific service requirements in a particular situation. Specificity is additionally increased through the individual orchestration of software components.
- *Characteristic of Uncertainty*: Implementation processes are generally exposed to varying degrees of uncertainty [15]. With the implementation processes of conventional ERP systems pre-defined sets of methods can be employed in order to reduce uncertainty and increase process quality [27]. Due to the uniqueness of a SOA based solution such pre-defined method sets can hardly be established [4].
- *Characteristic of Strategic Relevance*: The characteristics of strategic relevance address the importance of an implementation process for the participating actors. A high strategic relevance is given, if the implementation process is aimed at establishing a competitive advantage [24]. As service orientation is considered as an instrument of business process reengineering and business process improvement aimed at creating a competitive advantage, the task of implementing SOA as such is of high strategic relevance.
- *Characteristic of Frequency*: The frequency covers the number of subsequent transactions between the ERP adopter and the consulting company. The implementation of conventional ERP systems usually constitutes a single (even though long lasting) transaction which deploys the chosen system throughout the company [28]. As stated above, using SOA the implementation process is split up into many specific micro-implementation processes, therefore leading to a higher transaction frequency [29].

Referring to PAT, the most important characteristic of the principal agent relationships addressed in the following analysis is the unequal distribution of information between the two contracting parties.

A *first-best solution* would lead to an exchange of services without any transaction cost. This type of solution would be realized if information is available at no cost. However, the assumptions of Principal Agent Theory show that this is not the case in reality. Because of limited rationality and opportunistic behaviour the contractual parties are left with a *second best solution* caused by unequal distribution of information and a trade-off between residing uncertainness and the agency cost of signalling (on agent side) and monitoring (on principal side).

With rising uncertainty about the outcome of the contractual agreement the necessity for a monitoring effort on the principal side and the freedom of acting on the agent side increases. The principal's ability to realize if the result of the contractual agreement is reached because of the agent's behaviour is decreasing at the same time. In such situations he therefore cannot or only at a high cost,employ proper monitoring mechanisms. The evaluation of agency costs in market-based and hierarchical-oriented contractual relations is shown in the following figure.

Development of agency cost in dependency of relational situaion of contracting parties

Agency costs type	Market-based contractual relations	Hierarchical oriented contractual relations
Signalling	High	Low
Monitoring	High	Middle/Low
Welfare Losses	Ambiguous	Low

Fig. 2. Development of agency cost in dependency of relational situation of contracting parties

Applying the perspective of PAT the conclusion can therefore be stated as follows: the more specific the required information, the higher the agency cost of signalling and monitoring and the higher the potential loss in welfare because of non-optimal second best solutions will be. Following this conclusion the non-availability of highly specific information on both sides could furthermore be a *prohibitive factor* for contractual agreements (see Hold-Up problem) regarding the implementation of service oriented ERP systems in a market-based form of organization.

As we can see from our argumentation from a transaction costs perspective, the specificity of necessary information increases when moving from implementing conventional to service oriented ERP systems – as service oriented systems are implemented along business processes from a specific applying organization.

An efficient way to cope with this effect and reduce agency costs is the establishment of a long-term relationship between principal and agent. This can be most easily achieved when combining both parties in a hierarchical relation. Hierarchical relations enable easier monitoring and signalling and therefore enable efficient contractual agreements for the implementation process of service oriented ERP systems.

Following this argument our former conclusion of a shift towards hierarchical forms of organizing the implementation process of service oriented ERP systems is also supported from a Principal-Agent Theory perspective.

4 Organization of the Implementation Process

The choice of a suitable form of organization for the implementation process is determined by the actual combination of characteristic values that were presented in section 3. For syntheses, according to Transaction Cost Theory the relationships between characteristic values and transaction costs have to be considered.

As stated by WILLIAMSON the following generic recommendations for choosing an organizational arrangement for the implementation process can be derived [25]:

- The higher the estimate of characteristic values in a particular situation, the more beneficial it would be to integrate implementation service provider and implementation service customer by means of a hierarchy.
- Low characteristic values, however, suggest the utilization of the market in order to achieve the lowest transaction costs. In this case, service provider and customer conclude with a contract about expected services and results.
- Medium characteristic values suggest a choice for hybrid organizational forms in terms of a partnership or an alliance, enabling multiple agreements between service provider and customer.

Below, characteristic values for the two transaction types "implementation of conventional ERP systems "and "implementation of service oriented ERP systems" are illustrated (Fig. 3). The comparison of characteristic values may help to recommend a suitable organizational form for the undertaking of a transaction.

As proposed by Transaction Cost Theory, conventional ERP systems are optimally implemented by means of a hybrid relationship tending to a market-based approach (market- or hierarchical-based).

Drawing on Fig. 3 the specificity of the exchanged implementation services is higher than in a conventional implementation process. In conventional implementation

Comparison of characteristic values in an ERP implementation process

Characteristic	Conventional	Service-oriented
Specifitcity	Middle	High
Uncertainity	Middle/High	High
Strategic importance	High	High
Frequency	Low	Middle/High

Fig. 3. Comparison of characteristic values of ERP implementation transactions

processes the implementing company uses pre-defined blueprints that correspond to the customizable business processes of the chosen ERP system. However, considerable effort has to be put into customizing these processes of conventional ERP systems. In the case of service oriented ERP systems consulting companies cannot develop blueprints in advance since the business processes which are focused on, might vary for each implementation project. Furthermore, procedures for applying these blueprints within the SOA context are much unlikely to be reused, therefore limiting the possibility of regain the investment in developing them. Also specific investments on the side of the applying organization to cover the expertise needed are higher than in conventional implementation settings where they can be obtained on the market. This is because consulting companies can concentrate on one conventional ERP package and its implementation. However, they cannot provide expertise on the connection of all available software modules or services in the same way.

As stated above, uncertainty is a part of every implementation process. However, the level of uncertainty increases when implementing service oriented architectures. Applying organizations can rely on the expertise of consulting companies or other implementing partners when choosing to implement a well-known ERP package. They cannot expect this level of predefined processes and implementation know-how when a solution of individually packaged software modules should be installed.

The increase in uncertainty shows one important contribution of the Principal Agent Theory to our analysis. Uncertainty in that case is the main driver of an efficient design of organizational arrangements. Uncertainty refers to the difficulty of getting information about the contracting partner throughout the course of transactions. As consequence, agency costs rise. In order to reduce these additional costs, specific contractual arrangements could be bargained. However, the more complex these arrangements are the more inefficient the organizational arrangement due to burdening obligations for both principals and agents. Finally this results in additional welfare losses. The conclusion must therefore be to enable settings that reduce uncertainty and enable the parties to gain trustworthy information about each other. Long-term relationships as can be established in a hierarchical organizational setting enable such relationships.

The strategic importance of realizing a service oriented software solution is considered to be at a very high level. As there is a higher level of specific investments for the realization of a service oriented implementation process compared to conventional ERP system the risk of failure and the amount of investment that is connected with it is increased. The strategic importance can be considered to be high either due to the fact that companies only undertake the effort of service oriented implementation in areas where they consider their processes to contribute to a competitive advantage.

Service oriented ERP systems promise to deliver an increased flexibility compared with conventional ones. This means that they are more open to changes in business processes. The frequency with which an implementation process is executed is therefore increased by using SOA. On the opposite, conventional ERP systems are facing longer periods of stable architecture along their life-cycle.

Since all examined characteristic values for SOA are comparably high a vertical integration of transaction partners (i.e. within the applying organization or within a hybrid organizational form very similar to a firm) is – according to both Transaction Costs Theory and Principal-Agent Theory – suggested as an optimal organizational form. The necessary analysis of a firm's business processes and the considerable effort for selecting suitable application services requires a high degree of cooperation and control in order to successfully complete the implementation process. Contingent forms of cooperation are strategic alliances or long-term partnerships between the ERP adopters and consultancy firms. An integration of individual external specialists into a firm which are in charge of an implementation process is also a possibility in order to cover existing knowledge gaps.

5 Summary and Outlook

With this contribution we addressed the topic of implementing service oriented ERP systems from an organizational perspective. The comparison of conventional and service oriented ERP system implementation processes on the basis of New Institutional Economics suggests that implementation tasks in the context of service oriented ERP systems are better conducted within a hierarchical organizational arrangement. The overall implementation process of service oriented ERP systems is split up into many micro-implementation processes for single services and these micro-processes are much likely to occur frequently compared with conventional ERP system implementations. The practical implications are that the organizational arrangements between applying organization, consulting company and software vendors have to be reconsidered when implementing service oriented ERP systems. Also, the services offered by consultancies might have to be augmented. Independent consulting companies have to prepare for the demand of more specific knowledge in the field of business process analysis, therefore complementing their technical know-how. Furthermore, our results suggest that consultancies might embark on micro-implementation projects in the future characterized by a much shorter planning horizon. Consulting companies have therefore to specialize not only on the software side but also in the business process management practices relevant for the domains these consultancies are active in. With the applying organizations competences to analyze their business processes and align the ERP solution accordingly have to be built-up. The responsibility of the overall implementation project remains with the ERP adopter [30].

As our research is in an early stage, we only considered key concepts of the Transaction Costs Theory and Principal-Agency Theory respectively. We understand our analysis as a starting point into the field of implementing service oriented ERP systems. Future research should extend this limited scope, e.g. by considering additional organizational theories. A further limitation of our paper is the basic qualitative assessment of the implementation process based on the New Institutional Economics. Future research demand is therefore the extension of this assessment by means of empirical evidence and a sound quantitative analysis of relevant data. To this end we are striving to formalize the assessment of the implementation process by means of quantitative model.

However, first examples may already provide evidence for our suggestions. The increasing number of consultancy service providers which maintain close relationships with the applying organizations (e.g. Porsche Consulting, Siemens Consulting, Wuerth Consulting) can be regarded as a possible outcome of this development. In particular, business processes should constitute the focal point of any implementation process, therefore requiring an in-depth business process analysis by a service provider. Coordinators complement the firm's methodical knowledge about business processes and their implementation within ERP systems through the mediation of independent specialists for specific problems [30].

References

1. Themistocleous, M., Irani, Z., O'Keefe, R.M., Paul, J.R.: ERP and Application Integration. Business Process Management Journal 7(3), 195–204 (2001)
2. Gartner (July 30, 2008), http://www.gartner.com/it/page.jsp?id=728811
3. Rettig, C.: The Trouble with Enterprise Software. MIT Sloan Management Review, 20–27 (2007)
4. Schelp, J., Winter, R.: Enterprise Portals und Enterprise Application Integration. HMD Praxis der Wirtschaftsinformatik 39(225), 6–20 (2002)
5. Kajko-Mattsson, M., Lewis, G.A., Smith, D.B.: A Framework for Roles for Development, Evolution and Maintenance of SOA-Based Systems. In: International Workshop on Systems Development in SOA Environments (SDSOA 2007) (2007)
6. vom Brocke, J., Thomas, O., Sonnenberg, C., Schenk, B.: Economics of Service Oriented Architectures - Measuring the Financial Performance. In: Proceedings of the 7th International Conference on Perspectives in Business Information Research (BIR 2008), pp. 613–625 (2008)
7. Williamson, O.E.: The New Institutional Economics: Taking Stock, Looking Ahead. Journal of Economic Literature 38, 595–613 (2000)
8. Soja, P.: Success Factors in ERP Systems Implementations: Lessons from Practice. Journal of Enterprise Information Management 19(6), 646–661 (2006)
9. Finney, S., Corbett, M.: ERP implementation: a compilation and analysis of critical success factors. Business Process Management Journal 13(3), 329–347 (2007)
10. Ramayah, T., May-Chiun, L.: Impact of shared beliefs on "perceived usefulness" and "ease of use" in the implementation of an enterprise resource planning system. Management Research News 30(6), 420–431 (2007)
11. Peslak, A.R., Subramanian, G.H., Clayton, G.E.: The Phases of ERP Software implementation and maintenance: A Model for predicting preferred ERP use. The Journal of Computer Information Systems 48(2), 25–33 (2008)
12. Thamhain, H.J.: 15 Rules for Consulting in Support of a Client Project. Consulting to Management 15(2), 42–46 (2004)
13. Vathanophas, V.: Business Process Approach towards an inter-organizational Enterprise System. Business Process Management Journal 13(3), 433–450 (2007)
14. Parthasarathy, S., Anbazhagan, N.: Evaluating ERP Implementation Choices Using AHP. International Journal of Enterprise Information Systems 3(3), 52–65 (2007)
15. Kirchmer, M.: Business process oriented implementation of standard software. Springer, Berlin (1999)
16. Dai, Z., Duserick, F.: Toward Process-oriented ERP Implementation. Competition Forum 4(1), 148–158 (2006)
17. Scott, J., Vessey, I.: Implementing Enterprise Resource Planning Systems: The Role of Learning from Failure. Information Systems Frontiers 2(2), 213–232 (2000)
18. Kumar, K., Van Hillegersberg, J.: ERP-Experiences and evolution. Communications of the ACM 43(4), 22–26 (2000)

19. Nüttgens, M., Iskender, D.: Geschäftsmodelle für dienstebasierte Informationssysteme - Ein strategischer Ansatz zur Vermarktung von Webservices. Wirtschaftsinformatik 50(1), 31–38 (2008)
20. Zencke, P., Eichin, R.: SAP Business by Design - Die neue Mittelstandslösung der SAP. Wirtschaftsinformatik 1, 47–51 (2008)
21. Klesse, M., Wortmann, F., Schelp, J.: Erfolgsfaktoren der Applikationsintegration. Wirtschaftsinformatik 47(4), 259–267 (2005)
22. Scheer, A.W., Abolhassan, F., Jost, W.: Business Process Automation - ARIS in Practice. Springer, Berlin (2004)
23. Coase, R.H.: The Nature of the Firm. Economica 4(11), 386–405 (1937)
24. Kantarelis, D.: Theories of the Firm. Inderscience, Geneve (2007)
25. Williamson, O.E.: Comparative Economic Organization: The Analysis of Discrete Structural Alternatives. American Science Quarterly 36(2), 269–296 (1991)
26. Laffont, J.-J., Martimort, D.: Theory of Incentives: The Principal-agent Model. Princeton University Press, Princeton (2002)
27. Scheer, A.W., Köppen, A. (eds.): Consulting: Wissen für Strategie-, Prozess- und IT-Beratung. Springer, Berlin (2001)
28. Sahai, A., Graupner, S.: Web services in the Enterprise: Concepts Standards, Solutions, and Management. Springer, New York (2005)
29. Schelp, J.: Entwurf von Anwendungssystemen und Entwurf von Enterprise Services – Ähnlichkeiten und Unterschiede. Wirtschaftsinformatik 50(1), 6–15 (2008)
30. Schmidt, S.L., Vogt, P., Richter, A.: The Strategy Consulting Value Chain is breaking up. Consulting to Management 16(1), 39–44 (2005)

Author Index