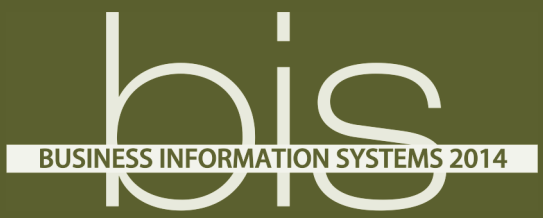
Witold Abramowicz
Angelika Kokkinaki (Eds.)

Business Information Systems

LNBIP 176

17th International Conference, BIS 2014
Larnaca, Cyprus, May 22–23, 2014
Proceedings



Springer

Lecture Notes
in Business Information Processing 176

Series Editors

Witold Abramowicz
Angelika Kokkinaki (Eds.)

Business Information Systems

17th International Conference, BIS 2014
Larnaca, Cyprus, May 22-23, 2014
Proceedings

Springer

Volume Editors

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

Angelika Kokkinaki
University of Nicosia
Department of Management and MIS
Engomi, Cyprus
E-mail: kokkinaki.a@unic.ac.cy

ISSN 1865-1348 e-ISSN 1865-1356
ISBN 978-3-319-06694-3 e-ISBN 978-3-319-06695-0
DOI 10.1007/978-3-319-06695-0
Springer Cham Heidelberg New York Dordrecht London

Library of Congress Control Number: 2014936884

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

Printed on acid-free paper

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

Preface

The 17th edition of International Conference on Business Information Systems was held in Larnaca, Cyprus. The BIS conference was established in 1997 and since then it grew to a well-renowned event of the scientific community. Every year researchers and business practitioners conduct scientific discussions on up-to-date issues that include the development, implementation, and application of business information systems. A wide scientific community and experts involved in the development of business computing applications participated in the 17th edition of BIS conference.

The BIS conference follows the trends in academic and business research; thus, the theme of the BIS 2014 conference was "Big data: problems solved and remaining challenges." The amount of data available for analysis, especially economy-related data, are increasing rapidly. According to MGI and McKinsey's Business Technology Group, big data is becoming a key basis of competition, an important contributing factor to productivity growth, innovation, consumer surplus as well as organizational effectiveness. Currently, big data is one of the most prominent trends in areas such as recommendation engines, network monitoring, sentiment analysis, fraud detection, risk modeling, marketing campaign analysis, and social graph analysis. However, there is a need to remodel technologies, applications, infrastructure, and business processes in order to take full advantage of the possibilities offered by big data.

In view of these, big data is becoming a very active area of research. Thus, the two first sessions of BIS 2014 were dedicated to relevant research. However, during the conference other research directions were also discussed, including business process management, ontologies and conceptual modeling, collaboration, service science and interoperability as well as specific BIS applications. The regular program was complemented by outstanding keynote speakers: Prof. George Giaglis (Athens University of Economics and Business, Greece), and Prof. Mike Papazoglou (University of Tilburg, The Netherlands) delivered enlightening and interesting speeches.

The Program Committee consisted of more than 90 members that carefully evaluated all the submitted papers. Based on their extensive reviews, a set of 22 papers were selected, grouped into seven sessions. We would like to thank the track chairs and reviewers for their time and effort. Their kind contribution was instrumental for the successful implementation of the conference. We are also grateful to the Organizing Committee co-chairs Elżbieta Bukowska (Poznań University of Economics) and Prof. George Papadopoulos (University of Cyprus) and the people who were involved in various aspects of organizing activities.

Finally, the significant contribution to BIS2014 of those who submitted their papers is deeply appreciated. Without their intellectual contributions, there would not have been a conference. We wish you all the best with your research endeavors and hope that the research included in these proceedings helps you reach your goals.

May 2014

Witold Abramowicz
Angelika Kokkinaki

Conference Organization

BIS 2014 was organized by Poznań University of Economics, Department of Information Systems, and University of Cyprus, Department of Computer Science.

Local Organization

Elżbieta Bukowska (Co-chair)	Poznań University of Economics, Poland
George Papadopoulos (Co-chair)	University of Cyprus, Cyprus
Barbara Gołębiewska	Poznań University of Economics, Poland
Włodzimierz Lewoniewski	Poznań University of Economics, Poland
Christos Metouris	University of Cyprus, Cyprus
Bartosz Perkowski	Poznań University of Economics, Poland
Wioletta Sokołowska	Poznań University of Economics, Poland
Milena Stróżyna	Poznań University of Economics, Poland

Program Committee

Dimitris Apostolou	University of Piraeus, Greece
Maurizio Atzori	University of Cagliari, Italy
Maria Bielikova	Slovak University of Technology in Bratislava, Slovakia
Michelangelo Ceci	Università degli Studi di Bari, Italy
Wojciech Cellary	Poznań University of Economics, Poland
Dickson K.W. Chiu	Dickson Computer Systems, Hong Kong, SAR China
Oscar Corcho	Universidad Politécnica de Madrid, Spain
Maria Luisa Damiani	University of Milan, Italy
Andrea De Lucia	University of Salerno, Italy
Tommaso Di Noia	Politecnico di Bari, Italy
Ciprian Dobre	University Politehnica of Bucharest, Romania
Josep Domingo-Ferrer	Universitat Rovira i Virgili, Spain
Schahram Dustdar	Vienna University of Technology, Austria
Suzanne Embury	University of Manchester, UK
Vadim Ermolayev	Zaporozhye National University, Ukraine
Anna Fensel	University of Innsbruck, Austria
Dieter Fensel	University of Innsbruck, Austria
Agata Filipowska	Poznań University of Economics, Poland
Adrian Florea	Lucian Blaga University of Sibiu, Romania
Vladimir Fomichov	Higher School of Economics, Russia

Ulrich Frank	University of Duisburg-Essen, Germany
Flavius Frasincar	Erasmus University Rotterdam, The Netherlands
Johann-Christoph Freytag	Humboldt Universität zu Berlin, Germany
Andrina Granić	University of Split, Croatia
Volker Gruhn	Universität Duisburg-Essen, Germany
Francesco Guerra	University of Modena, Italy
Hele-Mai Haav	Tallinn University of Technology, Estonia
Martin Hepp	Bundeswehr University of Munich, Germany
Knut Hinkelmann	University of Applied Sciences and Arts Northwestern Switzerland, Switzerland
Björn Johansson	Lund University, Sweden
Monika Kaczmarek	Poznań University of Economics, Poland
Pawel Kalczynski	California State University, USA
Kalinka Kaloyanova	University of Sofia, Bulgaria
Hariklea Kazeli	CYTA, Cyprus
Marite Kirikova	Riga Technical University, Latvia
Ralf Klischewski	German University in Cairo, Egypt
Jacek Kopecky	University of Portsmouth, UK
Mikail Kovalyov	National Academy of Sciences of Belarus, Belarus
Marek Kowalkiewicz	SAP Research, Australia
Helmut Krcmar	Technische Universität München, Germany
Dalia Kriksciuniene	Vilnius University, Lithuania
Maurizio Lenzerini	University of Rome La Sapienza, Italy
Peter Loos	Saarland University, Germany
Qiang Ma	Kyoto University, Japan
Maria Mach-Król	Katowice University of Economics, Poland
Leszek Maciaszek	Wrocław University of Economics, Poland
Panos Markopoulos	University of Nicosia, Cyprus
Florian Matthes	Technische Universität München, Germany
Heinrich C. Mayr	Alpen-Adria-Universität Klagenfurt, Austria
Nor Laila Md Noor	Universiti Teknologi MARA, Malaysia
Jan Mendling	Wirtschaftsuniversität Wien, Austria
Loizos Michael	Open University of Cyprus, Cyprus
Günter Müller	University of Freiburg, Germany
Markus Nüttgens	Universität Hamburg, Germany
Andreas Oberweis	Universität Karlsruhe, Germany
Mitsunori Ogihara	University of Miami, USA
Marcin Paprzycki	IBS PAN and WSM, Poland
Eric Paquet	National Research Council, Canada
Dana Petcu	West University of Timisoara, Romania

Elke Pulvermueller	University of Osnabrück, Germany
António Rito Silva	Instituto Superior Técnico, Portugal
Gustavo Rossi	National University of La Plata, Argentina
Massimo Ruffolo	University of Calabria, Italy
Virgilijus Sakalauskas	Vilnius University, Lithuania
Sherif Sakr	The University of New South Wales, Australia
Demetrios Sampson	University of Piraeus, Greece
Kurt Sandkuhl	University of Rostock, Germany
Juergen Sauer	University of Oldenburg, Germany
Ulf Seigerroth	Jönköping University, Sweden
Gheorghe Cosmin Silaghi	Babes-Bolyai University, Romania
Elmar J. Sinz	University of Bamberg, Germany
Athena Stassopoulou	University of Nicosia, Cyprus
Darijus Strasunskas	Norwegian University of Science and Technology, Norway
Jerzy Surma	Warsaw School of Economics, Poland
Miroslav Sveda	Brno University of Technology, Czech Republic
Dov Te'Eni	Tel Aviv University, Israel
Bernhard Thalheim	Christian Albrechts University of Kiel, Germany
Barbara Thoenssen	University of Applied Sciences Northwestern Switzerland, Switzerland
Ramayah Thurasamy	Universiti Sains Malaysia, Malaysia
Robert Tolksdorf	Freie Universität Berlin, Germany
Genny Tortora	University of Salerno, Italy
Wil van der Aalst	Technische Universiteit Eindhoven, The Netherlands
Vassilios Verykios	Hellenic Open University, Greece
Herna Viktor	University of Ottawa, Canada
Krzysztof Wecel	Poznań University of Economics, Poland
Mathias Weske	University of Potsdam, Germany
Anna Wingkvist	Linnaeus University, Sweden
Qi Yu	Rochester Institute of Technology, USA
Slawomir Zadrozny	Polish Academy of Sciences, Poland
John Zeleznikow	Victoria University, Australia
Jozef Zurada	University of Louisville, USA

Additional Reviewers

Al Machot, Fadi	Bock, Alexander
Andreou, Maria	Daeuble, Gerald
Apostolou, Dimitris	Dessi, Andrea
Batet, Montserrat	Dumont, Tobias
Baumgraß, Anne	Fasano, Fausto

Geuter, Juergen
Hajian, Sara
Harten, Clemens
Hauder, Matheus
Inzinger, Christian
Mueller-Wickop, Niels
Oro, Ermelinda
Ostuni, Vito Claudio
Otterbacher, Jahna
Pio, Gianvito
Reschenhofer, Thomas
Rogge-Solti, Andreas
Rogger, Michael
Rosica, Jessica
Scanniello, Giuseppe
Schauer, Carola
Schneider, Alexander W.
Thaler, Tom
Toma, Ioan
Vivas, José Luis
Walter, Jürgen
Werner, Michael

Table of Contents

Big Data

Business Process Management

Ontologies and Conceptual Modelling

Collaboration

Service Science and Interoperability

Specific BIS Applications

Optimizing Big Data Management Using Conceptual Graphs: A Mark-Based Approach

Yacine Djemaiel, Nejla Essaddi, and Noureddine Boudriga

Communication Networks and Security Research Lab.
Sup'Com University of Carthage, Tunisia
{ydjemaiel,s.nejla,noure.boudriga2}@gmail.com

Abstract. Nowadays, the optimization of the representation of big data and their retrieving is actually among the hot studied issues. In this context, this paper proposes a management scheme that enables the representation and the retrieve of big data, even if it is structured or not, based on extended conceptual graphs and the use of structured marks. A case study is given to illustrate the way to represent the generated big data needed to respond to distributed denial of service attacks according to the proposed management scheme and how the querying of such data may help to learn unknown attack fragments.

Keywords: big data, data marking, conceptual graph, data storage and retrieving, attack scenario.

1 Introduction

Nowadays, huge volumes of data are generated by running services for organization's information systems. Generated data may be structured, semi-structured or unstructured.

The concept big data has been defined as data that exceeds the capability of commonly used hardware environments and software tools to capture, manage, and process it within a tolerable elapsed time for its user population [2]. This concept is increasingly being defined by the 4 Vs, which are: 1) Volume which represents the size of the data; 2) Velocity that represents the rate at which data is being received and has to be acted upon is becoming much more real-time; 3) Variety which is considered for two aspects : syntax and semantics; and 4) Value that is especially attached to the commercial value any new sources and forms of data can add to the business.

The existing schemes are unable to retrieve all the data related to issued queries, since it is not represented in an efficient manner that helps to localize the needed data by reducing the size of the structures in addition to the retrieving processing time. In some cases, a structured query language is not applicable for big data since a significant part of the handled data can be unstructured. Moreover, the access time to the data to be retrieved may be affected by the important size of data. According to the available schemes, it is difficult to track changes to the data and to their associated security levels. In addition,

W. Abramowicz and A. Kokkinaki (Eds.): BIS 2014, LNBIP 176, pp. 1–12, 2014.

there is a lack of techniques that use big data through their querying systems in order to ensure protection regarding the security of a monitored system for example. Collected data from different sources may be related either due to the dependent services or the input/output relationships that may exist. However, these relationships are not preserved and considered by available management systems.

In order to deal with such shortcomings, this paper proposes a novel approach for representing big data and interrogating storage devices in an efficient manner making use of extended conceptual graphs in addition to the introduction of the mark concept as a structure used to define a set of attributes that help to localize the data and describe it in order to enhance the localization and the aggregation of the needed data as a response to a query. The contribution of this paper is four fold: 1) The definition of a novel management scheme for big data making use of new type of conceptual graphs; 2) The management of extended conceptual graphs to represent big data; 3) The proposal of a structured querying language that interrogates big data in an efficient manner by exploring conceptual graphs based on dynamic marks; and 4) The ability to trace inter-related data and learn new attack strategies based on the conceptual graph content, by considering mainly the marks.

The remaining of the paper is organized as follows. Section 2 presents the existing techniques for the management and the retrieving of big data. Section 3 introduces the principle for managing big data and the way to use the marking concept and details the proposed managing scheme. The next section introduces the novel defined querying language for big data by defining its syntax and the retrieving process as a response to a query. Section 5 illustrates the capabilities of the proposed approach for managing big data and enhancing the querying for a defined scenario dealing with the monitoring of enterprise information systems and business process against distributed denial of service attacks. The last section concludes the paper and discusses the possible enhancements to the proposed scheme in the near future.

2 State of the Art

Recently, a few approaches have been proposed for managing big data that are generated by the multiple running services in an information system.

Hadoop is among the frameworks proposed for storage and processing of large data sets. This framework consists of two major components. The first component is the Hadoop File System (HDFS) that is a highly scalable and portable file system for storing big data. The second component is Map-Reduce, which is a programming model for processing the data in parallel. It works by breaking the processing into two phases: the map phase and the reduce phase [7]. Despite the enhancement on the processing of data introduced by map reduce, it is not suitable for a real time processing especially when it is needed to handle huge volumes of streaming data.

Among the issues that are related to big data, one can consider how to classify these huge volumes of data. In [6], a distributed online data classification framework is proposed where data is gathered by distributed data sources and processed by a heterogeneous set of distributed learners which learn online, at run-time, how to classify the different data streams either by using their locally available classification functions or by helping each other classifying each other's data. This scheme provides a solution to classify data without providing the way to link them in order to enhance the querying and the retrieve of all interrelated data. Moreover, big data may be structured or not. Structured data is organized into entities that have a defined format, such as XML documents or database tables according to a particular predefined schema. Semi-structured data is used only as a guide to the structure of the data. Unstructured data does not have any particular internal structure such as plain text or image data [7]. In this case, querying unstructured data using structured querying languages is not possible.

In this context, querying big data may be performed using a set of keywords that describe the fetched data. In [4], an approach for extracting keyword queries from structured queries is introduced. This approach is based on the generation of query graph that captures the essence of the structured query, such as the object instances mentioned in the query, the attributes of these instances, and the associations between these instances. The keyword set is selected from the node and edge labels of the graph. The description of data using keywords has enabled the querying of unstructured data but it is insufficient to collect more information about the requested data such as the application that has generated it or the target represented by such data in addition to the inability to determine the additional data depending to it (e.g. data handled by the same user or belonging to the same attack scenario).

Another language called UnQL (Unstructured Query Language) is proposed in [1] for querying data organized as a rooted, edge-labeled graph. According to this scheme, relational data is represented as fixed depth trees, and on such trees, UnQL is equivalent to the relational algebra. This language provides additional capabilities to allow for selection and manipulation of complex document structures. It allows users to query data within documents themselves, instead of having to predefine each type of data and specific information contained within a document. The principles of UnQL will work on an ordered tree model, however it is not clear how they can be extended to an ordered graph model that is the most appropriate structure that may represent big data.

The browsing through large data set is among the difficult and time consuming tasks. In [8], a survey of algorithms needed to handle large data sets by illustrating the needed structures and methods implemented in order to deal with huge data volumes, is given. These algorithms are based on a set of techniques such as decision tree learning in order to enhance the efficiency and performance of computation. In addition, several key issues related to the security of big data are discussed especially for the way to ensure data integrity.

Using decision tree learning from a very large set of data may engender a very low processing. In [5], the survey presents the major issues related to decision

trees (e.g. incremental induction of decision trees) and training data sets in addition to the techniques to optimize these structures by constructing for example smaller pruned trees with maximum classification accuracy on training data or by using neural networks to enhance decision tree learning.

3 The Proposed Approach for Managing Big Data

The proposed approach for the management of the big data is built on the use of extended conceptual graphs and requires the definition of a novel structure that is called a mark. This mark helps to learn additional information on the stored data and enables the querying of structured or unstructured data using a proposed SQL like querying language.

3.1 The Use of Conceptual Graphs for Big Data

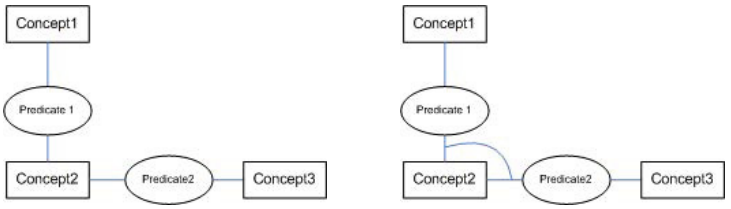
In order to enhance the retrieving of the required data as a response to a query, the data will be represented using a set of concepts and a set of relations between them. These concepts and the defined relations are added to the conceptual graph. Each generated mark associated to a data to be stored (text file, video, etc.) holds a set of concepts that are needed to build the conceptual graph.

Moreover, a structured SQL like query language is used to interrogate the CG in order to determine the required concepts. These concepts are attached to the remaining useful concepts in addition to the needed marks. Such marks are attached to the requested data. Therefore, the identified concepts as a response to the query lead to the needed data that will be given as a response to the query.

A conceptual graph is a finite, connected, bipartite graph composed of concepts and conceptual relations.

The basic components in the knowledge base are CGs, concepts, and conceptual relations. Using a Prolog-like notation, conceptual graphs are defined as predicates of the form, as defined in [3]: $cg(ID, RelationList)$ where ID is a unique identifier associated with this CG and $RelationList$ is a Prolog list that stores the conceptual relations of the specific CG. A conceptual relation is defined as: $cgr(RelationName, ConceptIDs)$ where $ConceptIDs$ is a list of concept identifiers linked by the specific conceptual relation. $RelationName$ is the name of the conceptual relation. Concepts are represented as predicates of the form: $cgc(ID, keywordList, Context, Concept\ Name)$ where ID is a unique identifier associated with this concept; $keywordList$ is a Prolog list of identifiers of the keywords containing this concept. $Context$ is either normal for the case of normal concepts or special for the case of marks; $ConceptName$ is the type-name of a normal concept or the mark label.

The predicates are used to define the relations between the different concepts. The operators $\wedge$ and $\vee$ are are used to represent relationships between the different concepts associated to a handled data. The first operator is used as an AND



(a) AND operator illustration (b) OR operator illustration

Fig. 1. Operators illustration

between predicates defined by $predicate_1$, $predicate_2$ linking concepts $concept_1$, $concept_2$ and $concept_3$, and represented as described by Figure 1a.

The relationship between the defined predicates using the OR operator is illustrated by Figure 1b.

3.2 The Mark Concept for Big Data

Data generated by the different available applications for an information system may be structured or unstructured with different types and formats. Each data may have some features and particularities that is known and managed by the application or the service that have generated such data. In order to deal with the data heterogeneity and variety, an additional structure should be generated and attached to the data. This structure is a mark that describes the data through a set of information that are related to the content such as : the user, the application that has generated the data, the identification of that data, a description of the content.

The mark is represented as a structure $m = \{id\_mark,\ source,\ target,\ user,\ id,\ concepts,\ predicates\}$, where the mark is identified by an id_mark. The *source* represents the application that has generated the data. The data represents the concept that is described by the *target* field. For each data, there is a user that has generated the data through an application. The *id* field holds the identification of the data associated to the mark. The field *concepts* composed of significant words that describe the data. The last field contains the set of predicates that defines the different relationships between the mark and the set of concepts in the CG that are associated to the data. An example of a mark is illustrated by the Figure 2 that defines the needed concepts and predicates that represents well the students data and helps to interrogate the defined text file and localize the needed data.

3.3 The Proposed Management Scheme

The proposed technique for the management of big data is inspired from the model proposed in [3] that is defined for video data. The proposed model is defined as follows:

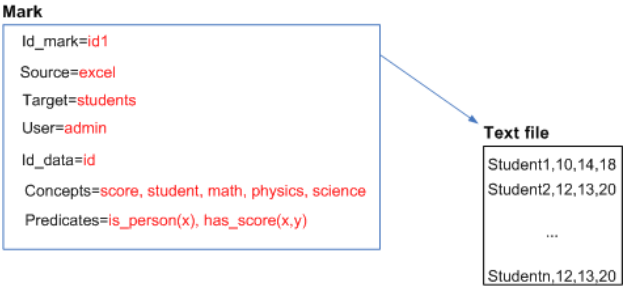


Fig. 2. Illustration of a mark and the associated data

$DM = (D, Map_1, M, Map_2, C, Map_3, KB)$ where D is the set of data to be handled that may be a file, a video, stream or a set of blocks in general that are denoted as $[b_i]$ where $i = 1, .., n$. The mapping relation Map_1 defines the correspondence between the marks and the data. This mapping is, in general, a many-to-many relationship since, in this way, a mark could be shared among data or a data (if it is fragmented for example) could have multiple marks (perhaps by different users). This mapping allows the identification of the needed data as a response to a query since the query language enables the identification of the marks. M represents the defined mark that holds the set of fields that are detailed in Section 3.2. Map_2 is a mapping relation that defines the relationships between a mark M and their concepts $C_1..C_n$. A concept may be attached to more than one mark. Map_2 is required to enhance the search and the retrieving process since it holds the association between the mark and the different concepts that represent the same data and that may be used as a criteria in the defined query by the user.

Map_3 is a mapping relation that maps a subset (application knowledge) of the KB to M since only this knowledge is directly derived from M. It holds the relationships that may exist between the different concepts belonging to the same mark.

An additional level of marking is introduced in order to enhance the response to a query by providing the fragment of data that is needed instead of the whole data (e.g. lines in a file). In order to achieve this goal, an additional level of marks is introduced by adding a mark for a fragment of data. In this case, a set of concepts are affected for a fragment of data and a relation is defined in this case. The definition of fragments depends on the used application and the type of data.

KB is the knowledge database that is encoded according to the CG knowledge representation scheme. It includes the different concepts, the existing relationships between them. The links between the concepts are defined using the set of available predicates.

The marking is performed by the application that generates the data based on the KB content associated to the handled data. The set of concepts and predicates are determined by querying the KB.

The generated graphs are stored in a distributed manner at each network that deploys the proposed management scheme for big data.

4 A Novel Structured Query Language for Big Data

The proposed novel structured query language for big data, called NSQL-DB is an SQL-like querying language that interrogates available marks and explores available conceptual graphs in order to determine the needed data even if it is structured or not. In this section, we introduce the syntax of NSQL-DB then the retrieving and updating processes according to such language, are explained.

4.1 Syntax

According to the NSQL-BD, the retrieving query is formulated using the following syntax :

select concepts where $(mark.field\ operator\ value \mid concept\ operator\ value \mid predicates)^{+}$.

The querying is performed based on a combination of several criteria including some mark fields values, concept values and predicates that are evaluated for concepts. $^{+}$ means that the query may include multiple criteria. The *operator* keyword represents the basic operators such as $=, \geqslant, \leqslant$, etc, in addition to the proposed relations such as Boolean (AND, OR and NOT), temporal adjacency (ADJ) and interval operators (DURING, OVERLAP, etc.). As a response to a query, a set of concepts are identified in the processed conceptual graphs that enables the identification of the related predicates and marks that are attached to the requested data.

The update of the conceptual graph content may be performed using the NSQL-BD by issuing a query according to the following syntax:

update mark|concept|predicate

set concept=value|mark.field=value|predicate=value

According to the aforementioned query, the update may concern all components of the conceptual graph including the marks, the concepts and predicates by affecting the defined values using the *set* keyword. If the specified criteria does not exist, the structure is added then the content is updated. For example, if the query specifies the following arguments for the *set* keyword : $mark.id = val$, $mark.id\_data = id$, $mark.concepts = file, session, hidden$ then the mark id is checked first, if it does not exist then a novel mark is created and the provided criteria are affected to the mark fields.

4.2 Retrieving Process

The fetching of the needed data as a response to a query is performed on the marks content instead of the huge volumes of stored data. The check is performed first for the set of mark groups. In order to enhance the retrieving process, a pre-processing is performed first in order to localize the set of group marks to be

fetched. This task is performed by considering the type of data that is required to be collected by considering several criteria including : the type of application that has generated the required data (for example a video application, a network analysis tool, etc.), the kind of data (e.g. text, video, binary, etc.), the target of the needed data (e.g. data related to a student, a product, a service, etc.), etc.

The first step is followed by the identification of sub marks as a response to the issued query. These sub marks help to identify sub conceptual graphs that are composed of the set of concepts and predicates which are related to the needed data or fragments of data. Figure 3 illustrates an example of the retrieving process for a sample partial graph and a sample query according to the proposed NSQL-BD language.

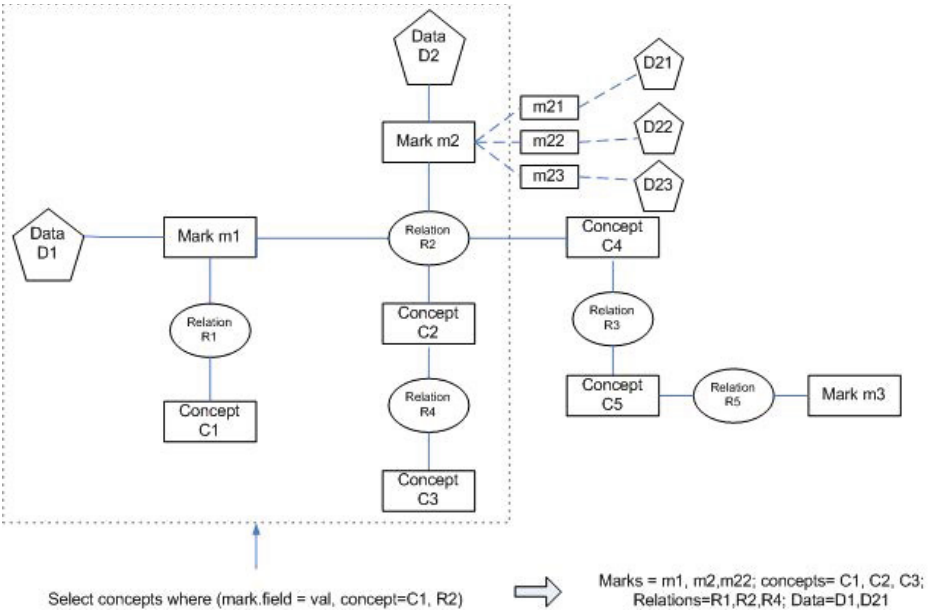


Fig. 3. Illustration of the retrieving process

4.3 Updating Process

The set of generated marks are regrouped in order to reduce the processing time for a query. The grouping is performed by checking the similarity rate that is calculated based on the different mark fields values. This operation is performed by checking for each mark the set of concepts that are related. Let define two marks m_i and m_j and their related set of concepts S_{m_i} and S_{m_j} where $S_{m_i}=\{c_{1_i}..c_{n_i}\}$ and $S_{m_j}=\{c_{1_j}..c_{m_j}\}$ m and n are their respective cardinalities. The two marks are regrouped if $S_{m_i} \cap S_{m_j} \neq \emptyset$. The result of grouping is a mark m_{ij} that has the following structure: id_group_mark,id_marks, $concepts$.

After regrouping the marks, a updated conceptual graph is generated. In this CG, the mark groups are considered as concepts and the dependencies between them are represented as relations if they exist.

An additional optimization should be performed on the generation of conceptual graphs and marks in order to prevent an explosion in the size of these graphs and a great increase of the number of generated marks. This optimization is based on a fusion of marks and the update of their content in terms of concepts and predicates. This task is performed by exploring conceptual graphs and by checking the similarity between the set of group marks considering their fields content. The check is performed in an incremental manner by increasing progressively the list of fields that are considered when interrogating the set of marks. If two marks m_i and m_j are considered similar, a new mark m_{ij} is generated considering the fusion of the content of both m_i and m_j and by eliminating duplicated data for mark field's values. As a consequence, their sub marks are also updated.

5 Case Study

Distributed denial of service attacks is among the security attacks that generate big data and may threaten the normal processing of enterprise information systems and business process. In this section, we present the way to represent such data according to the proposed management scheme and how the querying according to NSQL-BD helps to learn new attack scenarios including unknown malicious fragments.

5.1 Use of the Big Data Management System for the Monitoring and the Prevention of Advanced Security Attacks

The proposed management scheme uses conceptual graphs that holds the set of relations that are evaluated as predicates for stored data. The identification of the set of predicates that may be the starting fragment for an attack scenario is the response to NSQL-DB queries. The attack scenario may be seen as a set of predicates that are interrelated to each others using the defining operators and linking a set of concepts and their associated marks to the handled data. Dealing with security attacks, the set of predicates are regrouped as two categories: prerequisite predicates and consequence predicates. The former describes the set of predicates needed to initialize the attack against the information system components. The consequence predicates represent the set of evaluated concepts using a set of predicates that represent the effect of the attack on the handled resource of the information system. In most cases, a logical combination of predicates for complex attacks is defined. These combination are added to the mark and are represented through a graph. As an example, a combination of prerequisite predicates for a complex attack is as follows: $UDPVulnerableToBOF(VictimIP, VictimPort) \wedge UDP\ Accessible\ Via\ Firewall(VictimIP, Victim\ Port)$. This combination means that for a victim that is represented through

two concepts that are : $VictimIP$ and $VictimPort$, an attack aiming to gain access to a victim node may be initialized if the used UDP service is vulnerable to a buffer overflow attack and the firewall allows traffic targeting the UDP service to pass through. The proposed management system should be implemented by security solutions such as available intrusion detection systems, logging services at different levels (storage, system and network) that enables the generation of conceptual graphs associated to handled data and their attached marks. The data in this case are SYN packets targeting the victim in a SYN flooding attack which represent the big data in this case. These data are stored in persistent or temporarily manner in buffers, memory or storage devices. In order to prevent such attacks, it is required that the monitoring components are able to detect some fragments of predicates that belong to the running attack. It is possible also to follow a suspicious activity and to learn an attack scenario even if it holds a set of fragments that are unknown to available intrusion detection systems by exploring the available conceptual graphs through the NSQL-BD queries.

5.2 The Behavior of the Proposed Scheme

The attack strategy followed by the remote intruder is to compromise a great number of hosts belonging to the monitored site that implements the proposed management scheme for big data. He starts by scanning available hosts to verify if the vulnerable service is running, then he runs the exploit remotely. When he gains access to the system, he tries to prevent the download of the malicious scripts on the system, that may be detected by available intrusion systems. He creates the malicious file instead on the system and writes manually the code to be run by the script (e.g. a shell script) to use the compromised system as a zombie in this attack.

According to the proposed managing scheme for big data, the security monitoring system (e.g. IDS) checks generated conceptual graphs by fetching if there are fragments associated to a malicious activity that is represented as a set of concepts or predicates in the KB. Such KB contains a description of known attacks such as some types of scan attacks, as illustrated by Figure 4.

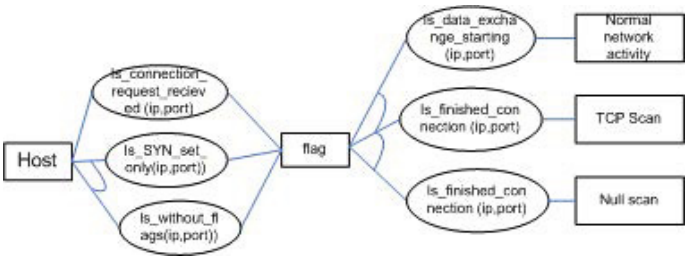


Fig. 4. Representation of some scan types in KB

The monitoring is ensured by issuing a set of queries that are defined using the NSQL-DB language. An example of a query processed by the monitoring system in order to detect malicious activity may be as follows: *select concepts where*$(is\_finished\_connection(*,*)\ AND\ concept = Null\ scan)$. $*$ means that all possible values for the required fields are considered (the arguments in this case are the ip address and the port). The partial generated conceptual graph related to the monitored activity, as a response to the performed query, is illustrated by Figure 5.

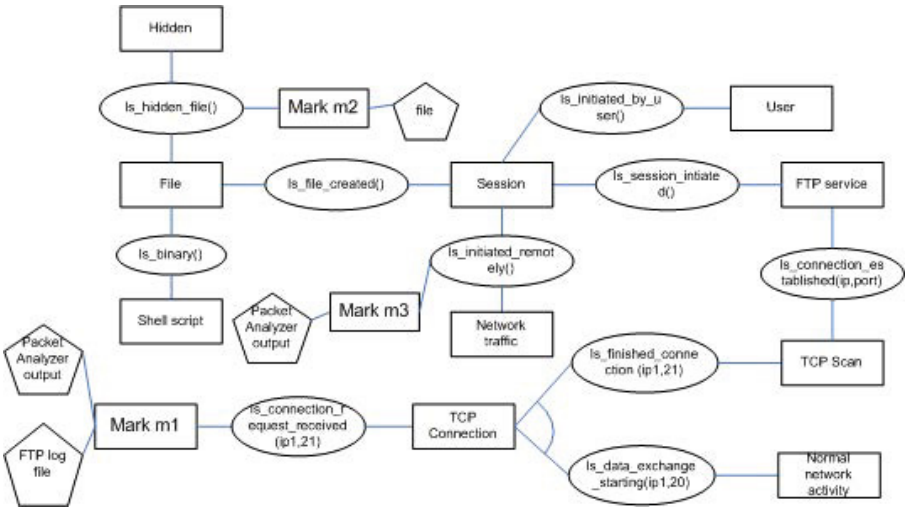


Fig. 5. Illustration of the partial generated CG

Among the generated marks, m_1 is associated to the monitored activity and attached to the captured traffic by the packet analyzer. This mark includes the following values for the different fields: $id\_mark = id$, $source = wireshark$, $target = TCP\ scan$, $user = admin$, id=$id\_packet\_analyzer\_output$, $concepts = TCP\ Connection, TCP\ Scan$, $predicates$= $is\_connection\_request\_received$ $(ip, port)$, $is\_finished\_connection(ip,port)$, $is\_data\_exchange\_starting$ $(ip, port)\}$. The partial conceptual graph enables the identification of the set of concepts and predicates that illustrate the different malicious actions and their related data including the log files, the captured network traffic, the network traffic stored temporarily in buffers in addition to the generated shell script that corresponds to the unknown attack fragment that has been linked to the performed attack.

6 Conclusion

In this paper, we have proposed a novel approach for managing big data that is built on the marking concept that attaches a set of metadata representing the

handled data either if it is structured or not. The proposed scheme ensures the generation and the management of a conceptual graph that includes the set of concepts that represent the stored data in addition to the existing relationships through the set of defined predicates that helps to identify the dependant data. The generated graph maintains in addition to these concepts the associations with the generated marks and their related data. Moreover, a querying language have been proposed to retrieve big data by exploring marks and concepts. The proposed querying language is SQL like and may be used to retrieve either structured or unstructured data. The efficiency of the proposed management scheme and the querying language are illustrated through a case study showing how to prevent distributed attacks by exploiting collected heterogeneous data. Several enhancements still possible for the proposed scheme mainly for defining strategies to optimize the content and the size of the generated conceptual graphs in addition to the way to store such structures and ensures their integrity and privacy.

References

1. Buneman, P., Davidson, S., Hillebrand, G., Suciu, D.: A query language and optimization techniques for unstructured data. SIGMOD Rec. 25(2), 505–516 (1996)
2. Franks, B.: Taming the big data tidal wave: Finding Opportunities in Huge data streams with advanced Analytics. Wiley (2012)
3. Kokkoras, F., Jiang, H., Vlahavas, I.P., Elmagarmid, A.K., Houstis, E.N., Aref, W.G.: Smart videotext: a video data model based on conceptual graphs. Multimedia Syst. 8(4), 328–338 (2002)
4. Liu, J., Dong, X., Halevy, A.Y.: Answering structured queries on unstructured data. In: WebDB (2006)
5. Patil, D.V., Bichkar, R.S.: Issues in optimization of decision tree learning: A survey. International Journal of Applied Information Systems 3(5), 13–29 (2012)
6. Tekin, C., van der Schaar, M.: Distributed online big data classification using context information. In: 51st Annual Allerton Conference on Communication, Control, and Computing (2013)
7. White, T.: Hadoop: The Definitive Guide, 1st edn. O'Reilly Media, Inc. (2009)
8. Yadav, C., Wang, S., Kumar, M.: Algorithm and approaches to handle large data-a survey. International Journal of Computer Science and Network 2(2) (2013)

DrugFusion - Retrieval Knowledge Management for Prediction of Adverse Drug Events

Mykola Galushka and Wasif Gilani

SAP (UK) Ltd,
The Concourse, Queen's Road,
Queen's Island,
Belfast, UK, BT3 9DT
{mykola.galushka,wasif.gilani}@sap.com

Abstract. This paper describes the highly scalable open source framework DrugFusion developed within the European Union(EU) project TIMBUS. It was designed by using Case-Based Reasoning(CBR) methodology to provide intelligent assistance for doctors and pharmacists in drug prescription process. DrugFusion analyses adverse event reports (AER)s published by United States Food and Drug Administration (FDA) and generates knowledge containers, which are used for efficient and accurate retrieval of similar treatment cases curried out in the past. DrugFusion uses the adaptation knowledge to produce a set of recommendations for medical practitioners, which allows them to make the most competent decision in planning a patient treatment. These recommendations include the most appropriate set of treatment drugs and warnings for the most likely adverse events. Considering the high complexity of the developed architecture, the main focus of this paper is on covering the similarity and indexing knowledge, used by the DrugFusion retrieval process.

Keywords: case-based reasoning, drug prescription, adverse drug events.

1 Introduction

On the European level exists a large body of secondary legislation regarding medical products for human and veterinarian use. Of great importance are e.g., the Directive on Medical Products for Human Use[1] or the Regulation on Advanced Therapy Medical Products[2]. While Directives must be transposed into national legislation of the Member States, Regulations are directly applicable.

Each prescription drug package selling in Europe must contain information about how it works and what the intended effect is. It must also contain description of side effects, instructions and cautions for its use, including warnings

[1] Directive 2001/83/EC of the European Parliament and of the Council of 6 November 2001 on the Community code relating to medicinal products for human use, O.J. L 2001/311, 67.

[2] Regulation (EC) No 1394/2007 of the European Parliament and of the Council of 13 November 2007 on advanced therapy medicinal products and amending Directive 2001/83/EC and Regulation (EC) No 726/2004, O.J. L 2007/324, 121.

W. Abramowicz and A. Kokkinaki (Eds.): BIS 2014, LNBIP 176, pp. 13–24, 2014.

about allergies. During a patient visit, a medical practitioner who is authorised to prescribe drugs, performs the patients initial assessment. During this assessment the medical practitioner tries to identify the best treatment strategy. It may include a prescription of one or more drugs, which need to be taken within the predefined time interval. Often prescribing drugs may cause adverse drug reaction (ADR) [3,4,13].

The study of ADRs is conducted in the field known as Pharmacovigilance. ADRs describe harms caused by taken medications at a normal dosage during normal use [5,2,1]. ADRs may occur in the following scenarios: a single dose, a prolonged usage of a drug or a result of combined use of two or more drugs (this scenario is specially targeted by DrugFusion CBR). ADRs expression has a different meaning than "side effect", since side effect might also imply that the effects can be beneficial. More general term, adverse drug event (ADE) [12,11], refers to any injury caused by the drug (whether drugs were used at normal dosage and/or due to overdose) and any harm associated with such case. It makes ADRs are a special type of ADEs.

Some companies have developed complex decision support systems, intending to help doctors and pharmacists to make the most effective choice of treatment with minimising the risk of occurring ADEs. The majority of these solutions are commercial, which prevent independent access to their algorithms and recommendation engines from a public domain. Considering the importance of issues related to the drug usage, the DrugFusion developed by the TIMBUS consortium [3] as a use-case, intends to provide public institutions with open platform for avoiding ADEs. The CBR system was selected as the main intelligent component for predicting potential ADEs. The general principals behind CBR approach and justification for it selection for DrugFusion are highlighted in the next section.

2 DrugFusion Architecture

CBR is a mature technology, which is based on a unique problem solving paradigm. This paradigm uses a psychological model of human problem solving. According to it, people often solve a new problem by reusing past experiences [9]. Here the current problem is solved by remembering a similar experience and re-using and adapting this old solution. In the same way, *CBR is able to utilize knowledge of previously experienced problem solving episodes (cases) in order to reuse them to solve new problems.* Such unique problem solving paradigm makes CBR an ideal choice to become the main intelligent component of the DrugFusion framework.The high level overview of the DrugFusion architecture is shown in figure 1.

As it can be seen in figure 1 the architectural components are associated with the corresponded CBR processes, described in the previous section: retrieval,

[3] This work has been funded by the TIMBUS project, co-funded by the European Union under the 7th Framework Programme for research and technological development and demonstration activities (FP7/2007-2013) under grant agreement no. 269940. The authors are solely responsible for the content of this paper.

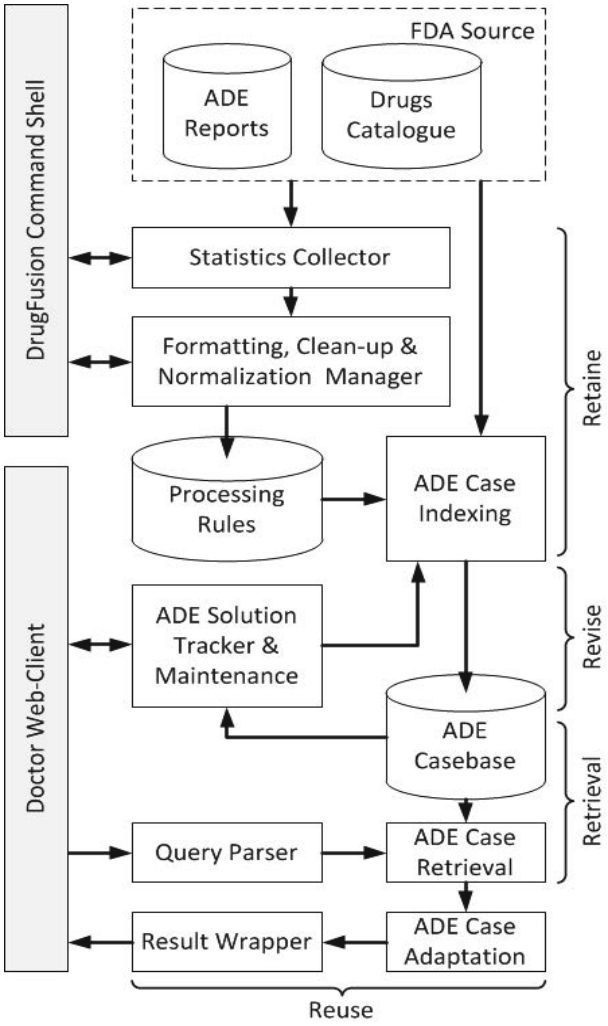


Fig. 1. DrugFusion Architecture

reuse, revise and retain. These processes are combined into the work-flow, which consists of three independent phases: *"Population"*, *"Retrieval and Adaptation"* and *"Validation and Maintenance"*. All these phases are described below.

2.1 Population

The population phase is initiated each quarter, when FDA publishes a new adverse events report (AER). DrugFusion monitors the FDA repository. When a new report becomes available, DrugFusion initiates its download and preprocessing. The downloaded report contains public information, such as subject

conditions, prescribed medications and occurred reactions, without any personal references, which can be used to identify a person.

A number of pre-processing steps are involved to address any inconsistencies within the input data. They mainly occur due to complexity of the data acquisition process and must be corrected before applying indexing algorithms.

DrugFusion performs three main pre-processing steps: *data formatting*, *clean-up* and *normalisation*.

- The data formatting phase analyses every row of the source dataset and identifies information entities. Rows can have a relatively high portion of errors, since their quality depends on the human effort of correctly specifying information about patient conditions and medication and occurred reactions. To overcome this problem DrugFusion uses an intelligent algorithm for identifying incorrect entities even in a highly obscured data input.
- The clean-up phase removes incomplete information. It scans each data row and validates each record against the predefined set of rules. Information about detected errors is stored in the clean-up report, where it can be evaluated by an expert. The expert is provided with a set of tools for correcting the isolated records and reinserting them into the initial dataset.
- The normalisation phase removes drugs spelling variations and maps various drug names to the unique identifiers. This procedure is very important for improving competency of the retrieval algorithm. Any inconsistencies within drug names will cloud the retrieval space, and prevent correct ranking of the search results.

Considering a possibility of a large number of reported events, the DrugFusion system administrator has a choice of selecting a file system for handling case-base. The first option allows to store the case-base under the local file system. This is a quick, inexpensive and easy way to manage the case repository. However, it has a number of disadvantages. With the incremental updates the case-base size may rapidly increase beyond a single disc capacity. Also data manipulation processes running within the local environment can significantly effect the overall system performance. To overcome these problems DrugFusion provides full integration with the distributed files system (HDFS), developed as a part of the Apache Hadoop project [10]. HDFS provides highly scalable case-base storage capacity with guaranteed data integrity. HDFS has also other advantages, such as allowing to deploy distributed approaches for case retrieval and adaptation which are presented in the next section.

2.2 Retrieval and Adaptation

The ADE search is initiated by a doctor or pharmacist as a part of the drug prescription process. The doctor uses a web-client provided by DrugFusion to specify a query, which may contain the following options: generic patient information, medical conditions, currently used drugs, observed ADEs and planned prescription details. The DrugFusion receives and analyses the specified query

and forms the target case. When this process has completed, the system retrieves similar treatment cases relevant to the specified target.

In the next step, DrugFusion performs an adaptation of the retrieved cases. The process of adaptation is a very complex task, which description lays beyond the scope of this paper.

Before the obtained solution is sent back to the doctor, the validation system performs an assessment of the identified ADEs. It compares all predicted ADEs with information provided by pharmaceutical companies for their products. Any inconsistencies between the obtained results during the adaptation process and description provided by manufacturers are immediately highlighted. This helps doctors to make a decision, whether to go ahead with the initial prescription options or to look for other alternatives. The final result set is wrapped into a format suitable for visualisation and sent back to the web-browser.

2.3 Validation and Maintenance

Validation and maintenance are the important feature for differentiating DrugFusion from other systems for prediction of ADEs. The overall DrugFusion intelligences not only depends on the AERS reports provided by FDA, but also on the internal case recycling process. This becomes one of the key advantages of using CBR as the intelligent core. Every time when doctor or pharmacist enters a choice of prescribed drugs, the system initiates the treatment tracking process over time. If a patient has developed an undesirable reaction, the outcome can be registered and the system will automatically modify its adaptation strategy ("learn from experience"). This self correcting approach leads to system learning, and as a result increases a competency of proposed solutions.

Such dynamic process of managing knowledge requires a powerful indexing structure, which not only provides efficient and accurate retrieval of similar cases, but also deals with utility problems. The utility problems are the serious issue in the development and maintenance of CBR systems. These are common for many problem-solving systems and reflect the situation when learning new knowledge affects a systems performance.

The decision was made to choose Discretised-Highest Similarity (D-HS) index [8] to make DrugFision system more resilient to utility problems and provide the efficient indexing and retrieval operations. The following sections describe the key points in applying this techniques for ADEs predictions.

3 Methodology

This section describes the D-HS indexing technique, which provides an efficient, competent and transparent indexing structure. It is suitable to the design of the retrieval knowledge container, which can be integrated into eHealth decision support systems for supporting drug prescription process. This section is split into tree parts. The case representation is described first. It is followed by

description of similarity metric used within D-HS. The last part describes the indexing structure, which provide efficient and accurate retrieval of similar cases for the specified target.

3.1 Case

The representation of the ADE case is shown in figure 2.

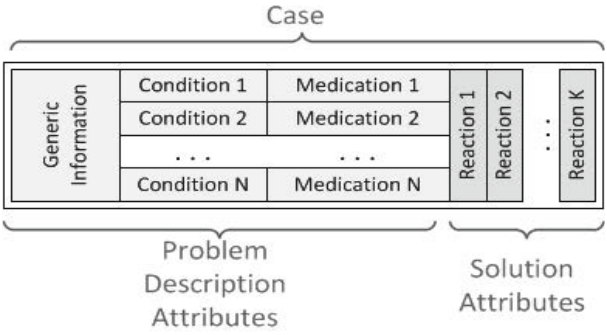


Fig. 2. A case representation

The case combines problem description attributes and solution attributes. The problem description attributes are used to identify case features, which are used during the indexing and retrieval processes. In DrugFusion the following problem description attributes are defined: a general information about a patient, such as age, gender etc., treatments (a combination of conditions and prescribed medications). The solution attributes are used to identify case features, which are used during the adaptation process. These features define adverse reactions. DrugFusion CBR handles conventional attribute types: *numeric, nominal and textual*. *Numeric attributes* are represented by real numbers. *Nominal attributes* are represented by discrete values and can be used to describe a counted subset of states. *Textual attributes* represent unstructured textual information and can be used to describe a particular item in natural language.

To simplify the methodology section lets combine all problem description attributes in one group and solution attributes in another.

3.2 Similarity

D-HS uses a predefined proximity area around the target case $\mathbf{q}$ which bounds a number of nearby cases. These cases are considered as closest by the D-HS similarity metric.

In D-HS, retrieval has the following definition (1):

$$\mathbf{C}'_{\text{D-HS}} = \mathbf{R}_{\text{D-HS}}(\mathbf{q}), \tag{1}$$

where $\mathbf{C'}_{\text{D-HS}}$ is the final result set of closest cases and where the function $\mathbf{R}_{\text{D-HS}}(\mathbf{q})$, performing the D-HS retrieval, is defined as (2):

$$\mathbf{R}_{\text{D-HS}}(\mathbf{q}) = \left\{ arg \max_{\mathbf{c} \in \mathbf{C}} dist(\mathbf{q}, \mathbf{c}) \right\}, \tag{2}$$

where the distance $dist(\mathbf{q}, \mathbf{c})$ is defined as (3):

$$dist(\mathbf{q}, \mathbf{c}) = \sum_{j=1}^{M} \begin{cases} 1, : \left(q_j - \dfrac{\Delta\delta}{2} \leqslant c_j \right) \wedge \left(c_j \leqslant q_j + \dfrac{\Delta\delta}{2} \right) \\ 0, : \left(q_j - \dfrac{\Delta\delta}{2} > c_j \right) \vee \left(c_j > q_j + \dfrac{\Delta\delta}{2} \right) \end{cases} . \tag{3}$$

A graphical representation for D-HS similarity determination is shown in Figure 3 (a) and (b).

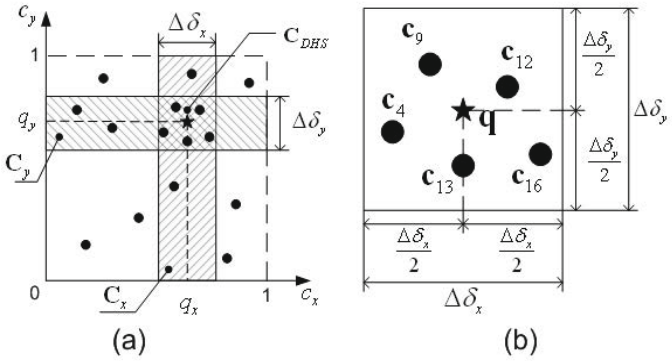


Fig. 3. (a) an example of the distribution of cases, the query case and a bounded area with closest cases; (b) a close up view of the bounded area with closest cases

It can be seen from Figure 3 (a) and definition (2), that the retrieval set $\mathbf{C'}_{\text{D-HS}}$ is obtained by the intersection of two initial retrieval sets $\mathbf{C}_x$ and $\mathbf{C}_y$, related to attributes c_x and c_y (4):

$$\mathbf{C'}_{\text{D-HS}} = \begin{cases} \mathbf{C}_x \bigcap \mathbf{C}_y \\ \mathbf{C}_x = \left(q_x - \dfrac{\Delta\delta}{2} \leqslant c_x \right) \wedge \left(c_x \leqslant q_x + \dfrac{\Delta\delta}{2} \right) \\ \mathbf{C}_y = \left(q_y - \dfrac{\Delta\delta}{2} \leqslant c_y \right) \wedge \left(c_y \leqslant q_y + \dfrac{\Delta\delta}{2} \right) \end{cases} . \tag{4}$$

The target case $\mathbf{q}$ is projected directly into the middle of the area $(\Delta\delta_x \times \Delta\delta_y)$ bounding the intersection set $\mathbf{C'}_{\text{D-HS}}$. The set $\mathbf{C}_x$ contains all cases falling into the interval $\left(\left(q_x - \dfrac{\Delta\delta}{2} \leqslant c_x \right) \wedge \left(c_x \leqslant q_x + \dfrac{\Delta\delta}{2} \right) \right)$ and $\mathbf{C}_y$ cases falling

into the interval $\left(\left(q_y - \frac{\Delta\delta}{2} \leqslant c_y\right) \wedge \left(c_y \leqslant q_y + \frac{\Delta\delta}{2}\right)\right)$ respectively. These intervals bound $\pm\Delta\delta$ areas around projections of the target case $\mathbf{q}$ attributes onto the corresponding dimensions. It is obvious, that it is impossible to predict in advance, the number of retrieved cases. In the worst case scenario it could be 0 (this situation is possible, but highly unlikely) or represent quite a high percentage of cases stored in the case-base, which inevitably would affect the competency of D-HS. The process of defining the "optimal" width of the interval $\Delta\delta$ is discussed in [7]. In Figure 3 (b), it can be seen that the result of the intersection of two sets $\mathbf{C}_x$ and $\mathbf{C}_y$, is the final retrieval set $\mathbf{C'}_{\text{D-HS}}$ consisting of five cases ($\mathbf{c}_4$, $\mathbf{c}_9$, $\mathbf{c}_{12}$, $\mathbf{c}_{13}$, $\mathbf{c}_{16}$).

In order to accurately define cases belonging to a particular interval, technically an exhaustive search should be carried out. The proposed indexing technique described in the next section, avoids such a routine search and makes the retrieval process more efficient.

3.3 Indexing

A grid type indexing structure as defined in (5) was proposed for constructing an index for the two dimensional space (for attributes: c_x and c_y):

$$\mathbf{R}_{\text{D-HS}}(\mathbf{q}) = \mathbf{C}^x_{l_x} \cap \mathbf{C}^y_{l_y}, \tag{5}$$

where $\mathbf{C}^x_{l_x}$ and $\mathbf{C}^y_{l_y}$ are containers (6) for attributes x and y, which include cases $\mathbf{c}_i$ lying within the same intervals l_x and l_y with the target case $\mathbf{q}$:

$$\begin{cases} \mathbf{C}^x_{l_x} = \{\mathbf{c}_i | \exists l_x : (0 \leqslant l_x \leqslant L-1) \wedge P(l_x, c_{ix}) \wedge P(l_x, q_x)\} \\ \mathbf{C}^y_{l_y} = \{\mathbf{c}_i | \exists l_y : (0 \leqslant l_y \leqslant L-1) \wedge P(l_y, c_{iy}) \wedge P(l_y, q_y)\} \\ i = \overline{1..N} \end{cases}, \tag{6}$$

where $P(l, x)$ is a predicate (7), which is *true* if the tested attribute x lies within the container with the index l and *false* otherwise:

$$P(l, x) = (l\Delta\delta \leqslant x) \wedge (x < (l+1)\Delta\delta)), \tag{7}$$

where the interval width is defined as $\Delta\delta = \frac{1}{L}$.

All dimensions for mapping case attributes are split into L intervals. Intervals have width $\Delta\delta = \frac{1}{L}$, and form containers $\mathbf{C}^x_{l_x}$ and $\mathbf{C}^y_{l_y}$, which amass all cases encompassed by them. $P(l, c)$ is a predicate, which is *true* when the attribute c falls in to the interval l and *false* otherwise. The process of indexing consists of projecting cases into the corresponding containers for each of their attributes. At the retrieval stage the technique identifies the corresponding container for each attribute of the target case, then extracts and intersects its contents. The result of this intersection is considered as a final retrieval set. The proposed indexing structure is shown in Figure 4.

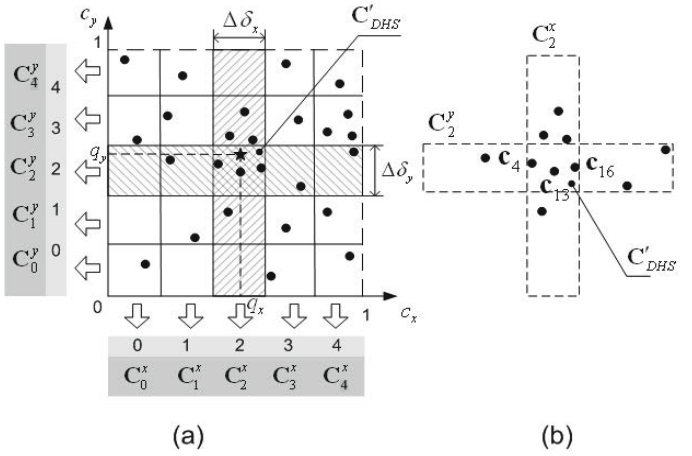


Fig. 4. (a) the distribution of cases within the grid, containers and the target case; (b) containers for target attributes q_x and q_y plus their contents

In Figure 4, the number of intervals is 5 ($L = 5$) for both attributes c_x and c_y . Therefore during the indexing process five containers ($\mathbf{C}_0^x$, $\mathbf{C}_1^x$, $\mathbf{C}_2^x$, $\mathbf{C}_3^x$, $\mathbf{C}_4^x$) are formed based on the following equal intervals [0, 0.2), [0.2, 0.4), [0.4, 0.6), [0.6, 0.8) and [0.8, 1.0] for the attribute c_x and five containers ($\mathbf{C}_0^y$, $\mathbf{C}_1^y$, $\mathbf{C}_2^y$, $\mathbf{C}_3^y$, $\mathbf{C}_4^y$) for the attribute c_y with the same intervals. The query case $\mathbf{q}$ falls into the interval with index 2 for the attribute c_x and into the interval 2 for the attribute c_y. Cases falling into these intervals are presented by the $\mathbf{C}_x^2$ and $\mathbf{C}_y^2$ containers. The intersection of these containers $\mathbf{C}_x^2 \bigcap \mathbf{C}_y^2$ produces the final retrieval set $\mathbf{C}'_{\text{D-HS}} = (\mathbf{c}_4, \mathbf{c}_{13}, \mathbf{c}_{16})$ for the specified query.

It can be seen from Figure 4 (a), that the position of the query case is not at the center of the area bounding the set $\mathbf{C}'_{\text{D-HS}}$. This is caused by using the predefined grid, for splitting cases into distinct containers based on their attribute values. As a consequence of this there is an increase in efficiency on the one hand and a possible decrease in competency on the other. The modifications which allow to improve competency of the D-HS algorithm are discussed in [7].

According to (5) the generalized definition of the D-HS retrieval structure for an n-dimensional space is (8):

$$\mathbf{R}_{\text{D-HS}}(\mathbf{q}) = \bigcap_{j=1}^{M} \mathbf{C}_{l_j}^j \tag{8}$$

where $\mathbf{C}_{l_j}^j$ is a container (9) for the attribute j, which includes cases $\mathbf{c}_i$ lying within the same intervals l_j with the target case $\mathbf{q}$:

$$\mathbf{C}_{l_j}^{j} = \{\mathbf{c}_i | \exists l_j : (0 \leqslant l_j \leqslant L-1) \wedge P(l_j, c_{ij}) \wedge P(l_j, q_j)\}, \tag{9}$$

where $i = \overline{1..N}$ and $P(l, x)$ is the predicate defined by (7) and $\Delta\delta = \frac{1}{L}$ is the interval width.

It can be seen from expression (8) that the D-HS retrieval is defined as the intersection of containers extracted for case attributes, where the D-HS indexing provides an infrastructure for matching those containers based on the query attribute values.

4 Experiments

The experimental case base is created by aggregating data presented in quarterly reports published by FDA. The performed experiments are carried out based on ASCII drug report. This report consists of nine files. Each file name has the following format: ¡file-descriptor¿yyQq.txt, where ¡file-descriptor¿ is a 4-letter abbreviation of the containing information type, 'yy' is a 2-digit representing the year, 'Q' is the letter Q, and 'q' is a 1-digit representing the quarter. (For example, 'DEMO96q1.txt' file represents demographic data type for 1st quarter of 1996.)

In all instances, the case-bases were split into training (9/10) and test (1/10) sets. Ten-fold cross validation was performed and the classification accuracy was noted for each tested technique along with the associated efficiencies. Efficiency is defined by natural logarithmic ratio of the time taken to complete 10 cross-validations between the tested technique (D-HS and C4.5) and the benchmark R-tree. The natural logarithm of the efficiency ratio was used to provide a clear and scaled visualisation of each technique's performance. A logarithmic value greater than 0 indicates that the technique is more efficient than the R-tree and a logarithmic value less than 0 indicates the converse. The significance in competency differences for each technique compared to the R-tree was calculated using paired t-tests (significance level p=0.05).

An R-tree was selected as the primary benchmark for comparing the efficiency and accuracy of the D-HS, due to its ability to perform k-NN queries [6] effectively by using the underlining tree-type indexing structure. It combines the k-NN accuracy on one hand and the tree-type index efficiency on the other. C4.5 was selected as an additional benchmark. The experiments were carried out on the case-base containing 100k reported ADEs. The evaluation results are shown in figure 5.

As it can be seen from the figure, the D-HS used by DrugFusion outperformed R-tree and C4.5 based solutions in terms of accuracy (78.2%) and efficiency (0.82). The D-HS accuracy result did not show a significant difference to R-tree. Despite small increase in accuracy (5.6%) D-HS showed the large improvement in efficiency (0.82) in comparison to both R-tree (benchmark) and C4.5 (0.34). The poor performance of the C4.5 can be explained by inability to come up with the clear problem space splitting strategy within the large number of predicting classes.

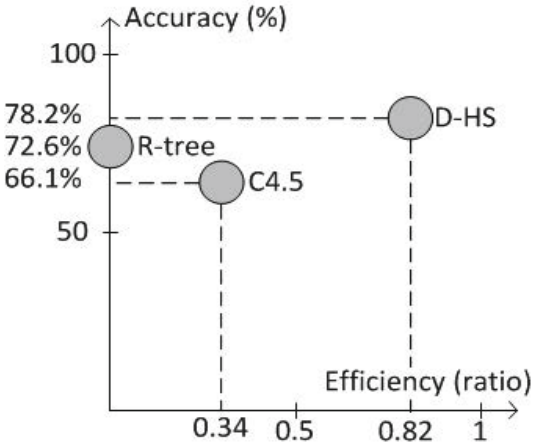


Fig. 5. Accuracy and efficiency of the DrugFusion ADEs prediction

5 Conclusion

DrugFusion is highly scalable, efficient and competent solution for prediction of adverse drug events, developed upon the case-based reasoning system. The high scalability and efficiency are achieved by implementing core functionality based on the Hadoop platform. The underlining HDFS allows efficient and reliable handling of the large volume of reported adverse events. The distributed computations constructed based on map-reduce approach allows the efficient data processing. The selected D-HS provides the straightforward parallel implementation for indexing and retrieval algorithms. It showed high competency, which authors plan to significantly improve by applying the D-HS modification for intelligent selection of indexing intervals. Considering complexity of queries in DrugFusion system, authors are planning to develop a graphical user interface for assisting in this process. Also an additional evaluation of DrugFusion system will be carried out to prove its efficiency and accuracy for the all FDA ADE reports.

Acknowledgment. The authors would like to thank all members of the TIM-BUS consortium, who provided the valuable feedback and recommendations during design and development of the DrugFusion system.

References

1. Ferner, R.E., McDowell, S.E.: Internet accounts of serious adverse drug reactions a study of experiences of stevens-johnson syndrome and toxic epidermal necrolysis. Can. J. Clin. Pharmacol. 8(2), 84–88 (2012)

2. Butt, T.F., Cox, A.R., Oyebode, J., Ferner, R.E.: Internet accounts of serious adverse drug reactions a study of experiences of stevens-johnson syndrome and toxic epidermal necrolysis. Drug Safety 35(12), 1159–1170 (2012)
3. Jin, H., Chen, J., He, H., Kelman, C., McAullay, D., O'Keefe, C.M.: Signaling potential adverse drug reactions from administrative health databases. IEEE Trans. Knowl. Data Eng. 22(6), 839–853 (2010)
4. Jin, H., Chen, J., He, H., Graham, W.J., Kelman, C., O'Keefe, C.M.: Mining unexpected temporal associations: Applications in detecting adverse drug reactions. IEEE Transactions on Information Technology in Biomedicine 12(4), 488–500 (2008)
5. Krska, J., Cox, A.R.: Adverse drug reactions. in: Clinical pharmacy and therapeutics. Clinical Pharmacology and Therapeutics 91, 467–474 (2012)
6. Kuan, J.K.P., Paul Fast, H.L.: k-nearest neighbour search for r-tree family. In: Proceedings on First International Conf. on Information, Communications, and Signal Processing, Singapore, pp. 924–928 (September 1997)
7. Galushka, M., Patterson, D.W., Nugent, C.: Assessment of four modifications of a novel indexing technique for case-based reasoning. Int. J. Intell. Syst. 22(4), 353–383 (2007)
8. Patterson, D.W., Rooney, N., Galushka, M.: Efficient similarity determination and case construction techniques for case-based reasoning. In: Craw, S., Preece, A.D. (eds.) ECCBR 2002. LNCS (LNAI), vol. 2416, pp. 292–305. Springer, Heidelberg (2002)
9. Schank, R.: Dynamic Memory: A Theory of Reminding and Learning in Computers and People. Cambridge University Press, New York (1983)
10. White, T.: Hadoop: The Definitive Guide. O'Reilly Media (2009)
11. Koutkias, V., Kilintzis, V., Stalidis, G., Lazou, K., Niès, J., Durand-Texte, L., McNair, P., Beuscart, R., Maglaveras, N.: Probability analysis on associations of adverse drug events with drug-drug interactions. BIBE 45(3), 1308–1312 (2007)
12. Koutkias, V., Kilintzis, V., Stalidis, G., Lazou, K., Niès, J., Durand-Texte, L., McNair, P., Beuscart, R., Maglaveras, N.: Knowledge engineering for adverse drug event prevention: On the design and development of a uniform, contextualized and sustainable knowledge-based framework. Journal of Biomedical Informatics 45(3), 495–506 (2012)
13. Ji, Y., Ying, H., Dews, P., Mansour, A., Tran, J., Miller, R.E., Massanari, R.M.: A potential causal association mining algorithm for screening adverse drug reactions in postmarketing surveillance. IEEE Transactions on Information Technology in Biomedicine 15(3), 428–437 (2011)

Prescriptive Analytics for Recommendation-Based Business Process Optimization

Christoph Gröger, Holger Schwarz, and Bernhard Mitschang

Institute of Parallel and Distributed Systems
University of Stuttgart Universitätsstr. 38, 70569 Stuttgart, Germany
{christoph.groeger,holger.schwarz,
bernhard.mitschang}@ipvs.uni-stuttgart.de

Abstract. Continuously improved business processes are a central success factor for companies. Yet, existing data analytics do not fully exploit the data generated during process execution. Particularly, they miss prescriptive techniques to transform analysis results into improvement actions. In this paper, we present the data-mining-driven concept of recommendation-based business process optimization on top of a holistic process warehouse. It prescriptively generates action recommendations during process execution to avoid a predicted metric deviation. We discuss data mining techniques and data structures for real-time prediction and recommendation generation and present a proof of concept based on a prototypical implementation in manufacturing.

Keywords: Prescriptive Analytics, Process Optimization, Process Warehouse, Data Mining, Business Intelligence, Decision Support.

1 Introduction

Today, adaptive and continuously improved business processes play a key role for companies to stay competitive. At this, the digitalization of process execution activities as well as the increasing use of sensor technologies lead to enormous amounts of data, from workflow execution data and machine data to quality data, posing a great potential for analytics-driven process improvement [1, 2].

Yet, existing process analytics in industry practice, e.g., as part of business activity monitoring approaches [3], do not fully exploit the valuable knowledge hidden in these huge amounts of data due to the following limitations: (1) they do not make use of prescriptive techniques to transform analysis results into concrete improvement actions leaving this step completely up to the subjective judgment of the user; (2) they do not integrate process data and operational data, e.g., from workflow management systems and enterprise resource planning systems, to take a holistic view on all process aspects; (3) the actual optimization is conducted ex-post after the completion of the process in contrast to a proactive improvement during process execution.

To address these issues, we present the data-mining-driven concept of *recommendation-based business process optimization (rBPO)* supporting adaptive

W. Abramowicz and A. Kokkinaki (Eds.): BIS 2014, LNBIP 176, pp. 25–37, 2014.

and continuously optimized business processes (see Fig. 1). rBPO exploits prescriptive analytics and proactively generates action recommendations during process execution in order to avoid a predicted metric deviation. It is based on a holistic process warehouse and employs classification techniques for real-time prediction and recommendation generation. For example, a worker is warned during process execution that the entire process is likely to run out of time, even if the current lead time meets the requirements. Then, a corresponding hint, e.g., to adjust a resource setting, is generated using data on past process executions in order to speed up processing and avoid the metric overrun. Thus, rBPO focuses on data-driven process optimization at runtime, not on classical process model improvement during design-time or ex-post analysis.

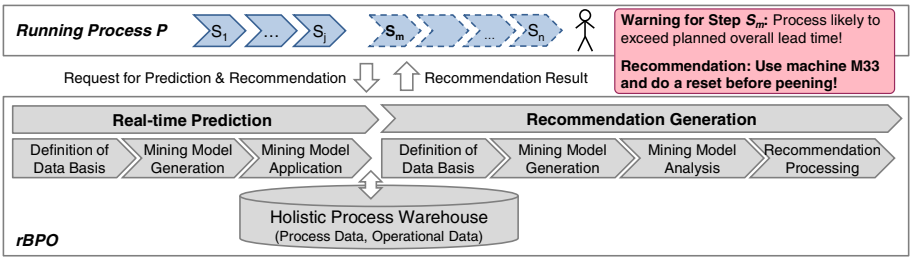


Fig. 1. Recommendation-based Business Process Optimization (rBPO)

The remainder of this paper is organized as follows: In Sec. 2, we define general requirements and present the basic approach of rBPO. Its two major components, real-time prediction and recommendation generation, are detailed in Sec. 3 and 4. Our proof of concept based on a prototypical implementation in the manufacturing industry is described in Sec. 5. Related work and a comparative evaluation of rBPO are discussed in Sec. 6. Finally, we conclude in Sec. 7 and highlight future work.

2 Requirements and Basic Approach

From a process management perspective, metrics are the basis for process optimization. The earlier potential metric deviations, e.g., excessive lead times, are detected during process execution, the more likely they can be avoided [4]. On this basis, we define the following core requirements (R_i) for rBPO.

The approach has to support a *metrics-based goal definition (R1)* and facilitate metric prediction and recommendation generation *proactively during process execution (R2)*. It should make use of all data generated across the entire business process in a holistic data basis. That is, it has to *integrate process data and operational data (R3)*. Process data is flow-oriented and comprises process execution data, i.e., process events, and process model data. Operational data is subject-oriented and provides additional information on process subjects like employees or machines [5]. Thereby, *recommendation generation should be adaptive (R4)* by exploiting the continuously growing data on completed process executions. Besides, recommendations should

comprise multiple actions (R5) in order to achieve the goal, e.g., by recommending both the specific resource to use and the corresponding resource settings.

To realize these requirements, rBPO comprises two major components, namely *real-time prediction* and *recommendation generation*, on top of a *holistic process warehouse (PWH)* (see Fig. 1). The PWH integrates process data and operational data across the entire process and additionally stores analysis results, e.g., data mining models, to enable their reuse.

It has to be remarked that rBPO is a universal concept which can be applied to different process domains like workflow-based processes or manufacturing processes as long as there exists a suitable PWH. In this paper, we focus on manufacturing processes because we developed a holistic process warehouse for manufacturing in our previous work [6]. As a running example throughout the paper, we refer to the manufacturing of steel springs in the automotive industry as described in [7]. It comprises amongst others steps for winding, tempering and shot peening of springs. Besides, an approach towards a holistic process warehouse for workflows can be found in [5] and may be used for rBPO, too.

To keep our approach generic, we only assume that a process P consists of n steps S_i with $P = \{S_1, \ldots, S_n\} \wedge 1 \leq i \leq n$. These steps may be executed not only in sequential but also in parallel and branched structures. Generally, rBPO can be applied to arbitrary process structures. For the sake of understandability, we refer to a sequential process in our examples where step S_{i+1} is executed directly after S_i.

Table 1. Data basis with process instance data including the running instance *i400*

Process Instance ID	Step1 Machine ID	Step1 Winding Speed	Step1 Empl ID	Step1 Empl Qualific	Step2 Machine ID	Step2 Tempering Temperature	Step3 Machine ID	Step3 Peening Reset	Step3 Peening Duration	...	Metric
i100	M12	120	E331	5	M23	290	M33	1	23		OK
i200	M12	135	E332	2	M23	291	M33	0	28		NotOK
i300	M12	135	E321	1	M23	290	M34	0	29		NotOK
i400	M12	121	E321	1	M23	285					
...			...	...							...

The *starting point of rBPO* is a holistic data basis with instance data about a process provided by the PWH in a denormalized structure (see Table 1). Each row comprises all data about all process steps of one instance of the process, e.g., details about machines settings. Moreover, the categorized value of the metric representing the optimization goal is added for completed process instances. For this purpose, the metric is selected in a preliminary step by an analyst who defines value ranges for undesired metric deviations. For example, lead times for the process of steel spring manufacturing which are higher than 27 minutes may be too high. This leads to a categorization of the metric with two values "OK" and "NotOK". It is important to remark that the metric refers to the entire process, not to a single process step.

For rBPO, a single instance of a process is analyzed during its execution according to the following two-step procedure, which is initiated every time a process step S_j with $j \in \{1, \ldots, n-1\}$ completes. Note that there is no need to run this procedure for the last step S_n in a sequential process.

(1) After the completion of step S_j, *real-time prediction* is run to forecast whether the entire process instance is likely to run into a metric deviation at the end, that is, whether the prediction reveals that the value for the metric will be "NotOK".

(2) If the latter is the case, there is a need for optimization to avoid the predicted metric deviation. Thus, *recommendation generation* is executed to deduce an action recommendation for the following process step S_m with $m = (j + 1) \leq n$.

In the following, we present an overview of real-time prediction in Sec. 3 and discuss recommendation generation as the central rBPO component in Sec. 4.

3 Real-Time Prediction

Real-time prediction comprises three steps, namely *definition of the data basis*, *mining model generation* and *mining model application* (see Fig. 1). In this section, we only highlight major technical aspects due to space limitations. A manufacturing-oriented discussion of real-time prediction issues can be found in our previous work [8]. With respect to the *definition of the data basis*, we refer to data about completed process instances in the PWH. From these process instances, we need (1) the attributes related to the already completed process steps of the running process instance for which we want to predict the metric value (*i400* in our example) and (2) the categorized metric value. In Table 1, this comprises the metric attribute and all attributes of steps one and two for process instances *i100* to *i300*.

For *mining model generation*, a suitable data mining technique has to be defined which uses the tailored data basis as training data. As we have to predict nominal values, classification techniques are employed [9]. Moreover, the generated mining model should be optionally presented to an expert user to enable him to comprehend the prediction and fine-tune parameters. Thus, the interpretability of the generated model should be comparatively high. In our previous work [8], we did a qualitative evaluation of major classification techniques with respect to their interpretability. To summarize, decision trees are comparably easy to understand and intuitively interpretable due to their graphical representation. Thus, we use decision tree induction as classification technique and focus on binary trees for the sake of enhanced understandability. The metric attribute represents the dependent attribute and the attributes of the set of completed steps $C_j = \{S_1, \ldots, S_j\}$ with $j \in \{1, \ldots, n-1\}$ are used as independent attributes for decision tree induction.

Finally, *mining model application* uses the decision tree to generate the prediction for the metric value. To this end, the data of the currently running process instance is used to traverse the decision tree and recommendation generation is started if the prediction reveals "NotOK".

4 Recommendation Generation

Recommendation generation deduces an action recommendation for the next process step in a running process instance. An action recommendation comprises several action items consisting of process attributes and a target value for each of them,

e.g., "Winding_Speed > 120". Thus, we base our concept on decision rules combining target values of process attributes. For this purpose, we generate decision trees which correlate the categorized metric value as a class label with selected attributes of selected process instances. Then, each path from the root node to a leaf node of the tree with the label "OK" represents a potential decision rule for a recommendation. We use decision trees to generate decision rules for the sake of comprehensibility for an expert user as described in Sec. 3. Another option could be to use association rule mining to deduce decision rules. Yet, this option lacks comprehensibility for the user. In addition, from a more technical point of view, since only association rules related to the metric attribute are relevant, computing *all* frequent item sets as commonly done in association rule discovery seems to be superfluous.

Recommendation generation encompasses the four sequential steps described in the following, namely *definition of the data basis, mining model generation, mining model analysis* and *recommendation processing* (see Fig. 1).

4.1 Definition of the Data Basis

The starting point for recommendation generation is the data basis provided by the PWH (see Table 1). In the following, restrictions on attributes and process instances used for recommendation generation for a process step S_m are discussed.

Regarding the *selection of attributes*, we generally assume that only attributes representing influencable factors like machine settings are considered, e. g., using a predefined filter. Thus, it is assured that recommendations only comprise directly applicable actions. Besides, attributes referring to completed process steps $C_j = \{S_1, \ldots, S_j\}$, are out of scope as they cannot be changed anymore. Moreover, attributes of all remaining process steps $R_m = \{S_m, \ldots, S_n\}$ with $m = j + 1 \leq n$ could be used to compare different recommendations for process step S_m with regard to their effects on later recommendations. Yet, this makes the evaluation of decision rules for the recommendation significantly more complex. Hence, we only use attributes of step S_m for recommendation generation for step S_m. In our example (see Table 1), these are amongst others "Machine_ID" and "Peening_Duration" for step S_3.

With respect to the *selection of process instances*, recommendations are derived using data about *completed* process instances because other running instances miss

Table 2. Data basis and restrictions for recommendation generation for process step S_3

Process Instance ID	Step1 Maschine ID	Step2 Material ID	Step2 Machine ID	Step3 Machine ID	Step3 Tool ID	Metric
i100	M12	MA43	M23	M33	T17	OK
i101	M12	MA43	M23	M33	T17	OK
i102	M12	MA43	M23	M33	T17	OK
i103	M12	MA44	M23	M34	**T18**	**NotOK**
i104	M12	MA44	M23	M34	T17	OK
i105	M12	MA44	M23	M34	**T18**	**NotOK**
i400	M12	MA44	M23			

Process restrictions:
- Material MA43 and Machine M33
- Material MA44 and Machine M34

the final metric value necessary for decision tree induction. Thereby, a decisive point is whether (1) all completed process instances are incorporated or whether (2) only completed instances which have the same attribute values as the currently running process instance are selected. To illustrate this point, Table 2 shows an exemplary data basis for recommendation generation for process step 3 in instance *i400*. Completed instances that have the same attribute value as the running instance are marked in blue (*i100-i105*). In general, there can be various implicit dependencies between process attributes due to process restrictions, which are not represented explicitly. For instance, certain materials used in step 2 may require specific machines in step 3.

In variant (1), one resulting recommendation based on the decision tree in Fig. 2 would be to use machine *M33* in step 3. Yet, this conflicts with the process restrictions because machine *M33* cannot be used with material *MA44*. In contrast, the decision tree in variant (2) reveals the one and only valid recommendation in this example, i.e., to use tool *T17*. This is because the dependency between material and machines can only be recognized by decision tree induction when solely instances *i103-i105* are used. Hence, we opt for variant (2), but we have to remark that a minimum amount of data about completed process instances with the same attribute values has to be available in order to recognize the dependencies.

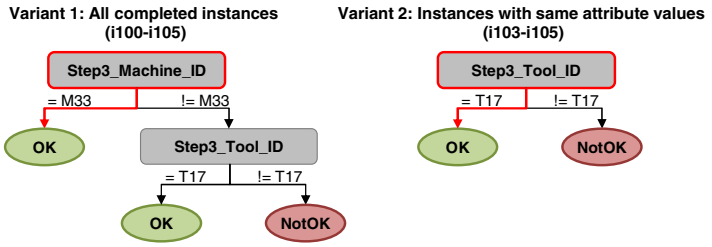


Fig. 2. Exemplary decision trees with different data bases

4.2 Mining Model Generation

Mining model generation comprises the generation of the decision tree on the data basis defined in Sec. 4.1. Below, we discuss the structure of the tree and its height.

With respect to the *structure of the decision tree*, we differentiate *binary trees* and *n-ary trees*, whereas the former has exactly two child nodes per node and the latter has arbitrarily many [9]. On the one hand, an *n-ary tree* reveals a *higher number of rules* due to the n-ary split, if we assume that a typical process attribute has more than two values. This increases the complexity of mining model analysis (see Sec. 4.3). On the other hand, the rules are supposed to be *more trustworthy* as they have a potentially *lower misclassification rate* compared to binary trees, when the maximum height of the trees is fixed. Yet, each decision rule in an n-ary tree is potentially backed up by *less underlying process instances* than a rule of a binary tree, if the maximum height is fixed. That is, the rules are derived from less process instances and thus are supposed to be *less significant*. Hence, for our approach, we opt for binary trees to

generate more significant recommendations and reduce the complexity of mining model analysis due to a smaller number of rules. Moreover, we decide to further restrict the trees by using only equal and not-equal relations for branches in order to speed up tree induction and simplify recommendations. That is, there are no subset restrictions on the branches. This makes recommendations more general and flexible. For instance, a recommendation may suggest employing all tools except tool 18.

With respect to the *height of the decision tree*, we define a maximum height depending on the concrete process and the number of process attributes in order to restrict the number of action items of a recommendation. Corresponding algorithms for decision tree induction and pruning to achieve the desired height are presented in Sec. 5.1.

4.3 Mining Model Analysis and Recommendation Processing

Based on the generated decision tree, mining model analysis comprises two steps, the *derivation of decision rules* and their *evaluation* in order to select the rule for the recommendation. For *rule derivation*, the tree is traversed from the root node to each leaf and each path which ends in a leaf node with the label "OK" results in a decision rule (see Fig. 3).

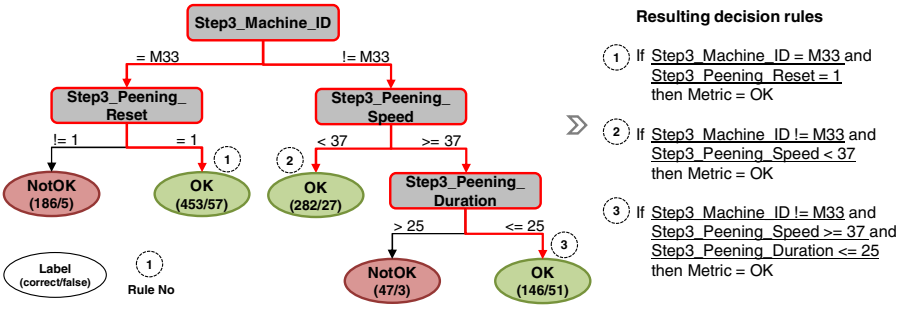


Fig. 3. Decision tree and resulting decision rules

Then, *rule evaluation* analyzes each rule according to the following four criteria:

The *misclassification rate q* is the percentage of process instances a rule does not classify correctly with respect to all instances covered by a rule, irrespective of their class label. The lower the misclassification rate, the higher is the trustworthiness of the rule. In general, rules are excluded that excess a threshold with, e.g., $q > 0.2$, depending on the concrete process.

The *percentage of underlying instances r* refers to the number of process instances a rule was derived from in relation to the total number of instances underlying the entire tree as training data. It represents the significance of the rule. Rules are excluded which do not apply to a minimum percentage of instances with, e.g., $r < 0.1$, depending on the concrete process.

The *length* of the rule *l* refers to the number of action items, that is, attribute-value-combinations of a rule. The shorter a rule, the easier it may be applied.

The *compliance with planned values c* refers to the correspondence of the action items of a rule with values defined during process planning, e.g., whether the recommended machine matches the planned one. Compliance *c* is defined as the percentage of matching attribute-value-combinations of a rule. The higher the compliance, the easier and faster a rule may be applied. Yet, it has to be remarked that there are attributes for which no production planning specifications are made and thus we do not define thresholds for *c*.

Table 3. Evaluation of decision rules

No	Rule	Missclassification (q)	Instances (r)	Length (l)	Compliance (c)	Score (t)
1	Step3_Machine_ID = M33 ∧ Step3_Peening_Reset = 1	11%	41%	2	50%	58
2	Step3_Machine_ID != M33 ∧ Step3_Peening_Speed < 37	9%	25%	2	0%	42
3	Step3_Machine_ID != M33 ∧ Step3_Peening_Speed >= 37 ∧ Step3_Peening_Duration <= 25	26%	16%	3	33%	-

For the final selection of a rule, we first filter all rules according to the defined thresholds for r and q in order to ensure a minimum quality of all rules. Then, we calculate a total score t with $0 \leq t \leq 100$ for each remaining rule and select the rule with the highest total score. To this end, a sub score is calculated for each criterion ranging from 0 to 25. These sub scores are summed up to the total score as follows:

$$t = ((1 - q) \times 25) + (r \times 25) + \left(\left(\frac{1}{l}\right) \times 25\right) + (c \times 25).$$

We equally weight all criteria but individual weights can be assigned depending on the concrete process. If there is more than one rule with the highest total score, they may be used alternatively or presented to the user. In the example (see Table 3), rule no 3 is excluded due to its excessive misclassification. Finally, total scores for rules no 1 and 2 are calculated and rule no 1 is selected for recommendation processing.

As the last step after mining model analysis, recommendation processing gets the selected decision rule as input and may either present it to the user in text form or feed it back in an operational system, e.g., a workflow management system, for further processing. For the sake of simplicity, we present the decision rule in human readable text to the user. In our example with rule no 1, a shop floor worker in manufacturing may receive the recommendation *"Use machine M33 and do a reset before peening"*.

5 Proof of Concept: Application in Manufacturing

Below, we provide a first technical proof of concept of rBPO based on a prototypical implementation for the manufacturing industry. Our *prototype* is based on the work of [10] and makes use of a relational implementation of the Manufacturing

Warehouse [6] as a PWH in IBM DB2. We use RapidMiner as a data mining tool and store decision trees in XML format. For decision tree induction, there is no generally valid algorithm as it heavily depends on the available data. For our prototype, we choose the C4.5 algorithm [11]. Alternatively, incremental decision tree algorithms [9] could be used to facilitate incremental model updating on the basis of new process instances. Real-time prediction and recommendation generation are implemented in Java.

For the *proof of concept*, we focus on the technical feasibility of rBPO and apply our prototype in an exemplary case, i.e., the manufacturing of steels springs for car motors as described in [7]. The process comprises sequential steps for winding, tempering, shot peening and testing of springs and employs different machines, e.g., winding automates and tempering furnaces. Based on a process study, we identified attributes of process steps and resources, e.g., winding speed and peening duration, as well as influence factors for metric deviations, e.g., machine settings. Then, we generated corresponding data on up to 100,000 process instances to populate the PWH.

On this basis, we *investigated recommendation generation* with respect to the requirements defined in Sec. 2. rBPO proved to proactively generate meaningful recommendations at process runtime (R2) focusing on metrics such as lead time and quality rate (R1). Thereby, it made use of operational and process manufacturing data (R3), e.g., from manufacturing execution systems and enterprise resource planning systems, integrated in the Manufacturing Warehouse. Decision trees were always generated on the entire data set to exploit the complete process history and realize adaptive recommendations (R4). The generated recommendations combined multiple action items (R5), e.g., on resources like machines, across different process steps.

In addition, we did *measurements* for multiple settings varying the number of process instances up to 100.000 instances on our test system (Windows Server 2008 R2, Core i7-2620M@2,7 GHz, 8 GB RAM). Each setting comprised 65 attributes across the process and was measured 5 times. For a setting with 100.000 process instances, our measurements reveal that data basis definition and mining model generation for real-time prediction take about 11 seconds on average. This is not critical as it can be done offline in advance. Mining model application takes less than one second which is suitable for online use at process runtime. For recommendation generation, data basis definition and mining model generation take less than 2 seconds on average. They have to be done online as the number of possible decision trees prevents an offline preparation. This is acceptable in typical manufacturing environments as there often is a delay between two manufacturing steps, e.g., due to transportation. Moreover, there is a significant potential for performance optimization, as the focus of our first proof of concept was on feasibility issues instead of pure response time. Finally, mining model analysis and textual presentation are done in less than 100 milliseconds.

To sum up, our first proof of concept demonstrates the technical feasibility and performance of rBPO. It proves that action recommendations can be proactively deduced at process runtime on the basis of a holistic process warehouse and that they can be generated quickly enough for an exemplary process environment. This provides the basis for an application in a real-world case in order to further evaluate the recommendations, e.g., comparing their effectiveness and comprehensibility.

6 Related Work and Evaluation

To structure related work, we differentiate three types of data analytics for process optimization [12]: *Descriptive analytics* focus on the manual and metrics-based analysis of completed processes as done in online analytical processing and reporting systems [13]. As opposed to that, *predictive analytics* forecast future process events. Recent approaches in process mining and business process intelligence [1, 14-17], e.g., for the prediction of metric values of running processes, belong to this category. Yet, all these approaches do not suggest concrete decisions but rather rely on the subjective judgment and analytical skills of the user to deduce improvement actions. In contrast, *prescriptive analytics* generate specific action recommendations to achieve a goal. That is, they build a bridge between pure analysis and actual optimization. In general, we observe two types of systems for prescriptive analytics: (1) recommender systems [18] using data mining techniques [9] and (2) expert systems [19] typically using rule-based, case-based and model-based reasoning techniques. We focus on data-mining-based concepts because expert systems require additional knowledge formalization and modeling and thus prevent a truly data-driven approach.

An initial approach towards prescriptive analytics for process optimization using decision trees is presented in [20]. It exploits a holistic process warehouse to generate decision trees predicting the performance of a new process instance. In case of a negative prediction, the instance is reconfigured before its execution. For the reconfiguration, the authors suggest to analyze the decision trees in order to deduce action recommendations. Yet, they do not provide any technical details on recommendation generation or decision tree evaluation and primarily focus on prediction issues.

Table 4. Comparative evaluation of rBPO

		rBPO	Pattern-based Optimization (PatOpt)	Recommendation-based Process Mining (RPM)	Risk-based Decision Support (RDS)
R1	Metrics-based Goal Definition	○	+	○	○
R2	Proactive Optimization during Process Execution	+	-	+	+
R3	Integration of Process Data and Operational Data	+	+	-	-
R4	Adaptive Recommendation Generation	+	-	+	+
R5	Multiple Action Recommendations	+	+	-	-

+ / ○ / **-** Approach fully/partially/does not meet(s) requirement.

To evaluate rBPO with respect to existing data-driven approaches, we did a qualitative comparison against the requirements defined in Sec. 2. For the comparison (see Table 4), we focus on the approaches of *pattern-based optimization (PatOpt)* [21], *recommendation generation using process mining techniques (RPM)* [22, 23] as well as the approach of *risk-based decision support (RDS)* [24] as these are the data-driven approaches most closely related to rBPO. PatOpt comprises a predefined catalogue of so called optimization patterns which encapsulate data mining techniques and

generate improvement recommendations, e.g., automating a certain decision activity to speed up the process. RPM focuses on operational decision support by recommending an action in order to optimize a metric, e.g., recommending the best resource for an activity. RDS predicts risks in terms of metrics deviations during process execution to provide decision support for certain actions, e.g., choosing the next process activity which minimizes process risks.

All four approaches support a *metrics-based goal definition (R1)*. Thereby, rBPO and RPM require the definition of one, possibly aggregated, target metric to be optimized, e.g., lead time. RDS focuses on risks in terms of metric deviations aggregated to a mathematical function. In contrast, PatOpt is based on the four target dimensions of process improvements, namely time, cost, quality and flexibility, and enables a multi-goal optimization. As opposed to the other approaches, PatOpt does not provide *proactive optimization (R2)* as optimization patterns are applied ex-post after process execution. With respect to the *data basis (R3)*, both rBPO and PatOpt are based on a holistic process warehouse integrating operational data and process data. RPM and RDS mainly focus on process data in an event log without explicitly integrating operational data on process subjects, e.g., machine data or master data on employees. Yet, we assume the integration of operational data to improve recommendation quality due to the augmented data basis. rBPO, RPM and RDS *adaptively generate recommendations (R4)* using data on past process executions. At this, rBPO employs classification techniques on warehouse data and RPM statistically evaluates the event log with traces of completed process instances. RDS employs decision tree induction on the event log to generate risk predictions. By contrast, the pattern catalogue of PatOpt constitutes a static collection of optimization best practices preventing adaptive recommendation generation. With respect to the generated recommendations, rBPO dynamically combines *multiple action items (R5)* of various types and PatOpt aggregates several optimization patterns. As opposed to that, RPM and RDS are less flexible and focus on a predefined type of action, e.g., to suggest the next activity to perform. At this, RDS provides only rudimentary decision support by predicting the potential risk for all possible actions without further concrete recommendations. All in all, rBPO goes beyond existing approaches by using a holistic data basis for adaptive recommendation generation in a fully data-driven manner.

7 Conclusion and Future Work

In this paper, we presented prescriptive analytics for recommendation-based business process optimization at process runtime. Our proof of concept underpins the technical feasibility and performance of our approach and emphasizes the importance of comprehensive data acquisition infrastructures, especially in manufacturing processes, to enable real-time process optimization. Moreover, it motivates the application in a real-world case to analyze recommendation quality and further refine decision rule evaluation. The realization of a closed loop feeding the recommendations back into a process control system is another aspect of future work.

References

1. Muehlen, M.Z., Shapiro, R.: Business Process Analytics. In: Vom Brocke, J., Rosemann, M. (eds.) Handbook on Business Process Management 2, pp. 137–158. Springer, Berlin (2010)
2. Kemper, H.-G., Baars, H., Lasi, H.: An Integrated Business Intelligence Framework. Closing the Gap Between IT Support for Management and for Production. In: Rausch, P., Sheta, A.F., Ayesh, A. (eds.) Business Intelligence and Performance Management, pp. 13–26. Springer, London (2013)
3. McCoy, D.W.: Business Activity Monitoring. Gartner Research Note (2002)
4. Melchert, F., Winter, R., Klesse, M.: Aligning Process Automation and Business Intelligence to Support Corporate Performance Management. In: Americas Conference on Information Systems (AMCIS), pp. 4053–4063. Assoc. f. Information Sys., New York (2004)
5. Radeschütz, S., Mitschang, B., Leymann, F.: Matching of Process Data and Operational Data for a Deep Business Analysis. In: Interoperability for Enterprise Software and Applications (IESA), pp. 171–182. Springer, Berlin (2008)
6. Gröger, C., Schlaudraff, J., Niedermann, F., Mitschang, B.: Warehousing Manufacturing Data. A Holistic Process Warehouse for Advanced Manufacturing Analytics. In: Cuzzocrea, A., Dayal, U. (eds.) DaWaK 2012. LNCS, vol. 7448, pp. 142–155. Springer, Heidelberg (2012)
7. Erlach, K.: Value stream design. The way to lean factory. Springer, Berlin (2011)
8. Gröger, C., Niedermann, F., Mitschang, B.: Data Mining-driven Manufacturing Process Optimization. In: World Congress on Engineering (WCE), pp. 1475–1481 (2012)
9. Han, J., Kamber, M., Pei, J.: Data Mining. Morgan Kaufmann, Waltham (2012)
10. Dapperheld, M.: Entwicklung analysebasierter Optimierungsmuster zur Verbesserung von Fertigungsprozessen. Master Thesis, University of Stuttgart (2013)
11. Quinlan, R.: C4.5: Programs for Machine Learning. Morgan Kaufmann, San Mateo (1993)
12. Evans, J.R., Lindner, C.H.: Business analytics. Decision Line 43, 4–6 (2012)
13. O'Brien, J.A., Marakas, G.M.: Management information systems. McGraw-Hill, New York (2011)
14. van der Aalst, W., Schonenberg, H., Song, M.: Time prediction based on process mining. Information Systems 36, 450–475 (2011)
15. 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, 239–273 (2004)
16. Grigori, D., Casati, F., Castellanos, M., Dayal, U., Sayal, M.S.M.: Business Process Intelligence. Computers in Industry 53, 321–343 (2004)
17. Kang, B., Lee, S.K., Min, Y.-B., Kang, S.-H., Cho, N.W.: Real-time Process Quality Control for Business Activity Monitoring. In: Computational Science and Its Applications (ICCSA), pp. 237–242. IEEE, Los Alamitos (2009)
18. Jannach, D., Zanker, M., Felfernig, A., Friedrich, G.: Recommender systems. Cambridge University Press, New York (2011)
19. Giarratano, J.C., Riley, G.: Expert systems. Thomson Course Technology, Boston (2005)
20. Grob, H.L., Bensberg, F., Coners, A.: Rule-based Control of Business Processes - A Process Mining Approach. Wirtschaftsinformatik 50, 268–281 (2008)
21. Niedermann, F., Radeschütz, S., Mitschang, B.: Business Process Optimization Using Formalized Optimization Patterns. In: Abramowicz, W. (ed.) BIS 2011. LNBIP, vol. 87, pp. 123–135. Springer, Heidelberg (2011)

22. Schonenberg, H., Weber, B., van Dongen, B.F., van der Aalst, W.M.P.: Supporting Flexible Processes through Recommendations Based on History. In: Dumas, M., Reichert, M., Shan, M.-C. (eds.) BPM 2008. LNCS, vol. 5240, pp. 51–66. Springer, Heidelberg (2008)
23. van der Aalst, W.M.P., Pesic, M., Song, M.: Beyond Process Mining: From the Past to Present and Future. In: Pernici, B. (ed.) CAiSE 2010. LNCS, vol. 6051, pp. 38–52. Springer, Heidelberg (2010)
24. Conforti, R., de Leoni, M., La Rosa, M., van der Aalst, W.M.P.: Supporting Risk-Informed Decisions during Business Process Execution. In: Salinesi, C., Norrie, M.C., Pastor, Ó. (eds.) CAiSE 2013. LNCS, vol. 7908, pp. 116–132. Springer, Heidelberg (2013)

Towards Planning and Control of Business Processes Based on Event-Based Predictions

Julian Krumeich[1], Sven Jacobi[2], Dirk Werth[1], and Peter Loos[1]

[1] German Research Center for Artificial Intelligence (DFKI),
Institute for Information Systems (IWi),
Stuhlsatzenhausweg 3, 66123 Saarbrücken, Germany
{Julian.Krumeich,Dirk.Werth,Peter.Loos}@dfki.de
[2] Saarstahl AG, Hofstattstr. 106, 66330 Völklingen, Germany
sven.jacobi@saarstahl.com

Abstract. To keep up with increasing market demands in global competition, companies are forced to dynamically adapt each of their business process executions to currently present business situations. Companies that are able to analyze the current state of their processes, forecast its most optimal progress and proactively control them based on reliable predictions, are a decisive step ahead competitors. The paper at hand exploits potentials of predictive analytics on big data aiming at event-based forecasts and proactive control of business processes. In doing so, the paper outlines—based on a case study of a steel producing company—which production-related data is currently collected forming a potential foundation for accurate forecasts. However, without dedicated methods of big data analytics, the sample company cannot utilize the potential of already available data for a proactive process control. Hence, the article forms a working and discussion basis for further research on big data analytics.

Keywords: Business process forecast and simulation, Predictive analytics, Complex event processing, Business process intelligence, Event-driven business process management, Business activity monitoring.

1 Introduction

1.1 Vision of the "Predictive Enterprise"

Global competition forces companies to increasingly individualize their business processes to meet the ever growing market requirements. A general view on business processes is no longer sufficient. Every single business process execution must be tailored to the present business situation. To achieve this in an efficient and timely manner, a high degree of automation is vital. The increasing digitalization of the real world, driven by the era of Internet of Things, allows for an unprecedented insight into current business process situations [1].

Companies that are able to analyze their business operations based on this rapidly growing mass of data, predict the best proceeding process flow, and proactively

W. Abramowicz and A. Kokkinaki (Eds.): BIS 2014, LNBIP 176, pp. 38–49, 2014.

control their processes with this knowledge are a substantial step ahead competitors. Such a company sketches the vision of a "Predictive Enterprise" as the next stage in the evolution of real-time enterprises within the age of data as a decisive competitive asset [2].

1.2 Motivation and Problem Definition

Today's corporate practice is far away from this vision. The potential of data, which is already collected in companies, has only received scant attention in business process management [3]. Frequently, this is due to the lack of technical capabilities to analyze these data in a timely manner and derive the right decisions. However, business analysts agree that companies are going to face existential difficulties, if problems in business process executions remain undiscovered or are not anticipated in time [4].

To counteract this, research needs to develop appropriate analytical methods and software systems that are able to detect and predict relevant events from collected data sets on time. The knowledge gained can then be used for a proactive control of business processes considering individual context situations.

Currently available enterprise software is able to process real-time process data—using Business Activity Monitoring (BAM)—and combined with complex event processing (CEP)—a technique to identify complex events—it can handle complex business events for deriving aggregated information [1]. However, existing predictive analytic approaches, such as Business Process Intelligence (BPI), rarely take into account data that is near real-time nor do they sufficiently use technologies like CEP to consider the current context situation appropriately. Existing algorithms are simply not capable to analyze data that is already available in many companies in a satisfactory time frame. Nevertheless, the available data sets will keep running up calling for dedicated big data analytics solution that exploit the full potential of such data. Innovative enterprise software therefore needs to come up with intelligent process control capabilities taking this large volume of data as a promising foundation for accurate event-based process planning and control.

1.3 Research Contribution and Applied Research Method

The paper at hand proposes the concept of event-based process predictions and outlines its potentials for planning, forecasting and eventually controlling of business processes (see sec. 2). In doing so, it is sketched how the application of such event-based predictions can make manufacturing processes more efficient. After proposing the general concept of event-based predictions, the paper analyze by means of a case study (see sec. 3), which process and context data a sample steel producing company can currently collect by its applied sensor technology forming a potential basis for event-based predictions of their manufacturing processes.

Hence, the paper follows case study research, which has been applied in information systems research for almost two decades [5]. In particular, the methodology proposed by Benbasat et al. [6] had been employed: as the unit of analysis the steel bar production line at Saarstahl AG, a major German steel producer, was chosen. The two

research questions applied to the case study were “What type of data is currently available in industry processes using state of the art sensor technology to realize event-based predictions?” and “Why is it a ‘Big Data’ challenge to analyse this data appropriately?” The results of the selected case study are considered to be generalizable to other manufacturing enterprises. Since the chosen research design is a single-case study, it is particular appropriate to revelatory cases, which is typical for Big Data analytics cases as a relatively new phenomenon in information systems research. The case study data was collected and analyzed by the central department of information and communication technology of the company. As data selection methods, interview techniques were applied and physical artefact were investigated.

Based on the examined data, the paper concludes that with current state of the art (see sec. 4), the available data cannot be analyzed in a reasonable time horizon in order to make sufficient business value out of it. Without dedicated big data analytics, the sample company cannot exploit the full potential of event-based predictions based on its data. For this, the paper aims at forming a working and discussion basis for further research efforts in big data analytics (see sec. 5).

2 Planning and Control of Manufacturing Processes Using Event-Based Predictions

2.1 Basic Concept to Realize Event-Based Predictions

In enterprise resource planning (ERP) systems, forecast-based methods have been applied for years to determine production demands [7]. While independent requirements are usually calculated based on bills of materials, stochastic calculations are applied to determine dependent requirements. A forecast-based control of manufacturing processes is not yet existent. This is due to the fact that for deriving forecasts a certain inaccuracy is always accepted, since datasets required for the predictions are in most cases too small to cover the operational context situation sufficiently. While in principle, it is possible to expand the data base, such an information gathering proves to be too complex, too expensive and not accomplishable in a timely manner. Therefore the independent requirements planning and hence the higher-level production planning and control are deterministically handled so far. This is proved by current market studies. At present, companies still perform insufficient analyzes and forecasts based on their collected sensor data, although this could have an immense impact on their economic and ecological performance [3].

In the course of technological developments, especially in the fields of "Internet of Things" and "cyber-physical systems", it is possible to accumulate production processes relatively cost-neutral by various types of sensors. This allows to measure internal and external parameters of production processes in an arbitrarily fine granularity, even in real-time. Thereby the technical foundation is created to establish and continuously enrich a data base, so that predictions can also be used for critical planning purposes.

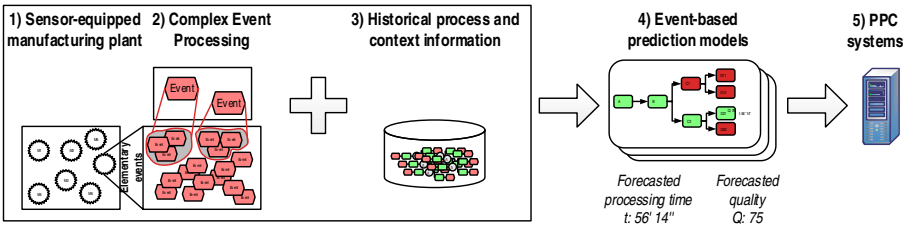


Fig. 1. Basic concept to derive event-based prediction

From the obtained mass of such process parameters, which can be regarded as elementary events, it is essential to identify patterns that can be utilized for process predictions (Fig. 1, 1). Here the research and technology field of CEP turns out to be promising. CEP has already been successfully used for fraud detection in banking and finance, in which financial transaction processes can be examined on a very fine-grained level by means of information technology [8]. Thus, with CEP, complex event patterns of a current process instance can be determined (see Fig. 1, 2) and by correlating these real-time patters with historical ones, which are linked with contextual information, event-based prediction models can eventually be derived (see Fig. 1, 3). These models allow to determine forecasts of substantially higher accuracy, since predictions are not purely based on stochastic, but instead, the actual current state is decisively considered for computation (see Fig. 1, 4). Hence, they can even be used for high-level production planning and control (PPC) systems (see Fig. 1, 5).

2.2 Example Scenario of Utilizing Event-Based Predictions in Manufacturing

The example below demonstrates the disadvantages of stochastic forecasts, outlines the potential of event-based predictions through utilizing CEP techniques and illustrates how they can be realized.

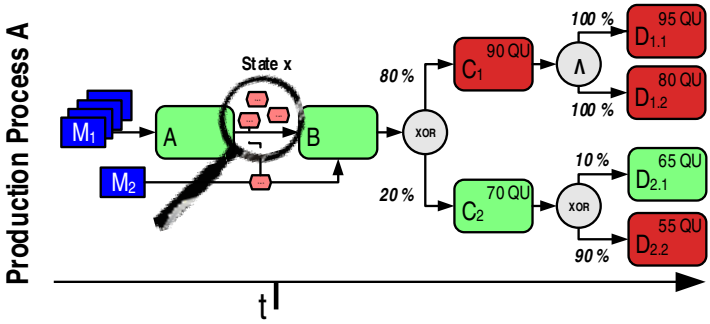


Fig. 2. Process sequence and forecast of production process A

A steel manufacturing company receives two customer orders for which a production planning is carried out. For the first order, a final product D with a minimum quality of 90 quality units (QU) has to be manufactured; the second order only

requires a minimum quality of 70 QU of product D. Based on the statistical frequency of occurrence of certain quality characteristics after passing through a manufacturing process A, classical stochastic forecasting approaches will determine an 80% probability that the final products of type D, have a quality of 95 QU (D1.1) or 80 QU (D1.2; see Fig. 2). Thus, both customer orders could be satisfied. In this context as the process proceeds, the forecast assumes further processing of the intermediate C, which is created with a quality of 90 QU. The timing of customer orders and the machine allocation plan would be based on this forecasted model.

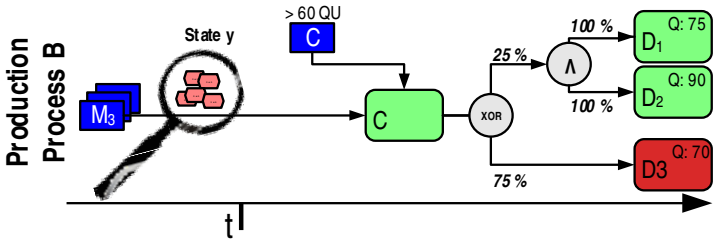


Fig. 3. Process sequence and forecast of production process B

In addition to manufacturing process A, manufacturing process B allows to produce the final product D out of intermediate products C (see Fig. 3). This manufacturing variant is particularly suitable if the resulting product C (in process A) does not have the required quality of at least 90 QU to serve the further processing to D > 80 QU.

However, this alternative processing consumes more time and a higher amount of material. Moreover, the end products of this production process meet only in 25% of all cases the required quality criteria. In the stochastically more likely case (75%) only one product with 70 QU can be manufactured and eventually be sold (see the lower process path in Fig. 3). In the worst case, this final product must therefore be disposed separately or, alternatively, needs to be returned to manufacturing process A (e.g., in steelmaking by melting it down again).

Thus, the use of such a purely stochastic forecasts would lead to the process execution outlined above with a probability of 80% (visualized in red). However, this means at the same time that this prediction is simply wrong in almost a quarter of all cases and would lead to final products that are not available without additional expense (cf. the lower process path in Fig. 2). Thus, the use of such purely stochastic forecasts proves to be insufficient, since it can only determine a forecast with minor significance and lead to an incorrect production planning.

If the considered manufacturing processes are equipped with appropriate sensors, a database can be developed that maps the diverse states of production and the corresponding manufacturing context patterns. This underlying data could be correlated with a current process state determined by CEP. Hence, in the outlined case a significantly more accurate prediction could be identified by having knowledge about states x and y. This is possible, since the forecast is not purely stochastic, but event-based.

This means for the underlying application example: if for instance after completion of process step A within manufacturing process A, a certain quality of the input raw material M_2, a machine variance of the machines used by va_1, ..., va_n, as well as the participation of employee m in process step B are detected, a complex event pattern will be identified that is correlated with the historical data base for a prediction. Based on the recognition of this complex event, the forecast would either significantly strengthen the probability of process variant A or, in contrast, predict the stochastic exception (see green process steps in Fig. 2). Considering the latter, the intermediate C with a quality of 70 QU would result with high significance. For its further processing to product D, a final quality of 65 QU is determined (see green process steps in Fig. 2). According to this, the computation forecasts that the customer orders cannot be satisfied as initially planned. Based on the determined values for C_2 and other process parameters, in accordance to the diagnosed state y, it will also be predicted that the further processing by manufacturing process B will significantly contribute to the stochastic unlikely final product D that is of sufficient quality to satisfy the customer orders (see green process steps shown in Fig. 3).

This shows that by incorporating current process and context parameters, the further course of a process can be predicted with considerably higher precision. If the outlined states x and y are recognized at a time t by CEP techniques, i.e., already recognized at the beginning of the production, the production could be planned more precisely and proactively controlled based on event-based predictions. For instance, certain setup procedures for starting manufacturing process B could already be performed in parallel to the running process steps B and C_2 in production process A. Stretched production time, increased material requirements as well as personnel and equipment utilization could be scheduled earlier or examined for the computation of possible alternatives.

While a stochastic forecast detection is quite simple to compute, event-based prognoses yet turn out to be extremely complex even for the simplified example. The individual process and context parameters form a highly interwoven construct, which complicates the correlation of individual basic events. Looking at the amount of data to be analyzed, it becomes evident that this is a challenge that must be tackled with innovative methods of big data analytics. A case study at Saarstahl AG, a large steel producing company, outlined in the next section brings out this fact by analyzing actual available data in a selected part of their production processes.

3 Case Study: Steel Bar Production at Saarstahl AG

3.1 Brief Description

This case study analyzes a production part within the steel bar production at Saarstahl AG, a major German steel producer, and provides types and sizes of sensor data that is currently collectable throughout the production processes. The outlined data was collected and evaluated by the central department of information and communication technology at Saarstahl AG. In the considered production area, half a million tons of

steel are produced annually. In order to meet customers' specific quality requirements for various existing end products, Saarstahl AG conducts comprehensive quality checks within its production line. These include diameter controls with laser, surface testing by magnetic particle testing, checks for internal steel errors by ultrasound, and a variety of temperature and vibration measurements. All of these techniques provide continuous sensor data at the lowest system level (L1). In addition, other sensor systems in production (ambient and positioning sensors) are installed to monitor the control of steel bars within the production line via a material flow tracking system (L2-system level). Based on this basic data and the available customer orders, the production planning and control system calculates a rough schedule (L3 to L4 system level).

3.2 Current State of Production Planning and Control at Saarstahl AG

Steelmaking processes are significantly influenced by internal and external events that lead to deviations in production quality. Some examples are fluctuating material properties of pig iron or production-induced deviations due to physical processes taking place. This entails a need for frequent adaptations of individual production instances within production and subsequently extensive re-planning of underlying production plans. For instance, if a standard further processing of steel provides the steps peeling, heat treatment and finally sawing of steel bars, then the crude steel cannot pass these steps if the sensor network used in the process detects a bending, as an example of a production deviation.

In this case, this intermediate either needs to be post-processed in a dressing and straightening machine to comply with quality requirements, or the processing of the intermediate can continue for another customer with lower quality criteria. Therefore the production planning for this process instance is obsolete and needs to be re-initiated in order to meet quality promises and deadlines to all customers, e.g., steel bars need to be factored into the normal process flow after conducting such an ad hoc processing step. Since the production planning systems compute an almost full capacity for the following days in batch mode, the simple insertion of this process instance into the running production flow may contradict to the planned execution of the production plan. Therefore, in case a production deviation is detected during the production process, an intelligent recalculation of the entire production plan would be necessary in real-time to ensure an optimal solution from a global planning perspective. If such production influencing events were even anticipatable, corresponding countermeasures could be initiated before development.

Currently, the data provided by the sensor network, which is integrated into the production processes, exceeds the volume that is analyzable so far. The information and control systems used and the analysis techniques available on the market do not allow the producer to monitor and control the entire production process in real-time. Moreover, no future states and events, such as looming production deviations, can be predicted on time. Hence, so far the control of production processes is rather done in a reactive way; yet, the sketched vision of a predictive enterprise has not been realized.

3.3 Data Characterization and Challenge of "Big Data"

In the following, sample data obtained from the applied sensor networks at Saarstahl AG are described according to the Big Data characteristics proposed by Gartner [9]. If the entire sensor networks within the production process is considered—which would be necessary for a comprehensive production planning and control on an L3-/L4-systems' level—, the "Big Data challenge" will be many times higher.

An example from the sensor network illustrates the immense *volume* of data in monitoring the production process. In the rolling mills 31 and 32 there are two optical surface test sensors that can continuously provide real-time data for the detection of surface defects during the rolling process. Basically, this allows to take into account the varying customer demands for a particular surface quality. The unit can already prototypically detect errors and differentiate the types of errors. This optical testing generates several hundred terabytes of data annually. Currently, only a sporadic reactive analysis of these data is possible. Also, other context data from the sensor network and the systems settled on L-2 or L-3 level can currently not be linked due to the volume of data to be analyzed. While these systems could in principle detect production deviations in batch mode, this takes too long to allow timely reactions.

While this is just one example of very large resulting data of individual sensors in a particular section of the factory, another example illustrates the high data diversity (*variety*), which is continuously collected by different sensors and sensor networks at various points throughout the production process. This places high demands on an analysis by big data principles. For instance, the further processing of steel bars as of now already provides half a million of sensor data records, which reflect one production area to a particular context. In the next couple of months the sensor performance will be advanced, such that over 1.5 million sensor data will be available on L1 and L2 level. According to the principles of CEP; however, only the identification of relevant events in this torrent of both homogeneous and heterogeneous data as well as their correlation allows to derive patterns and deviations. This is possible only by using highly structured, technical knowledge. At this point, the basic claim to a scalable solution becomes clear. Since such a sensor network should be flexibly expandable, but also allow analyses and forecasts within a required time frame. For instance, the company plans to increase sensing in this subsection to an output of more than five million records, which underlines the need for scalability.

Thus, in terms of the analysis of these large and diverse data, the responding time is crucial, since speed is a decisive competitive factor in the analysis (*Velocity / Analytics*). Classic reporting or batch processing would be significantly too slow, so that so-called high velocity technologies must be performed in near-real-time analyses. For the purposes of the outlined vision of predictive enterprises, it is also crucial to conduct accurate forecasts of the process sequences. Each day, an average of one terabyte of video data is recorded in a single subsection of the plant. However, a pure video analysis method is not sufficient for predictive analytics methodologies. In the existing system, it has been shown that only some production deviations could be detected by this classical approach. In addition, there is no feedback for process optimization. Therefore process data need to be included in the model formation and

forecasting. Here, as outlined, over one million data sets are incurred in the coming months. For analyzing the dependencies among process and video data, data from a long period of time must be used for model training. In this case, the data volume may rapidly exceed 50 terabytes. For a real-time adaptive prediction, on average one-tenth of the data should be used. At present, however, such a number of data can hardly be processed in real-time. A direct compression of the data is impossible because of its variety to be considered.

As of now, due to its big data characteristics, the production process is away from the envisioned optimum in terms of planning and control. Technically, Saarstahl AG can integrate further sensor technology into its production processes to achieve a better monitoring. However to increase the demanded business value, current analytical methods take too long to complete an analysis.

3.4 Required Technology to Address the Big Data Challenge

To address the outlined challenge and realize the vision sketched in sec. 1.1, intelligent predictive analytics systems are needed

- with the ability to analyze a variety of occurring (complex) events and data streams, which are obtained by means of integrated sensor networks in a real-time,
- which, for this purpose, filter irrelevant data and aggregate relevant one to have a more accurate picture of the current production state as provided by CEP, and
- that moreover are able to calculate predictions to prevent potentially conflict loaded situations in future and proactively adjust production planning.

The use of such an intelligent information system would allow Saarstahl AG to increase its operational as well as economic performance and be able to face global competition even more effectively.

4 State of the Art Discussion

4.1 Process Data and Context Data Acquisition and Its Real-Time Analysis

In order to detect context situations and progress of ongoing processes, data must be continuously collected. Technically, this can be done by means of physical and virtual sensors, which are often connected in a wireless network. Physical sensors are able to measure for instance pressure, temperature or vibration and recognise complex structures such as audio signals [10]. Additional data is obtained from IT systems by evaluating for example log files or exchanged messages at the lowest system level [11].

The next step is to analyse this mass of collected data. A common approach is that atomic events, which are detected by sensors, are being first cumulated to more complex ones, which is called CEP. CEP combines methods, techniques and tools to analyze mass of events in real-time and to obtain higher aggregated information from this most elementary data. Such approaches can be found for example in Wu et al. [12], who apply CEP on real-time data streams, and with particular respect to

RFID-based data streams in Wang et al. [13]. Often ontologies are used to identify semi-automatically the semantic context of events [14]. Operations on data of atomic and complex events require algorithms that are well-adapted to the process context as well as on big data characteristics.

A well-known approach for big data analysis is MapReduce, in which a problem is decomposed into independent sub-problems that are distributed and solved by so-called mappers. However, traditional forms of MapReduce follow a batch approach, which is inapplicable for data streams. In this regard, the field of research for stream mining and stream analytics has been formed recently [15].

4.2 Process Forecast Calculation and Simulation

Predictive analytics refers to the forecast of prospectively occurring events. The necessary mathematical modelling can be done by using historical data. A characteristic example of a simple forecast calculation is the moving average. Such simple models only work if they are mostly independent of external influencing events, which is rarely the case. Especially in business processes several dependencies and influencing factors exist. Hence, modern statistical methods are necessary to recognize dependencies and patterns in large amounts of data, such as decision trees, univariate and multivariate statistics as well as data warehouse algorithms [15].

Approaches combining predictive analytics with business process management, are summarized under the term Business Process Intelligence (BPI) (cf. [16] for a discussion on existing definitions). Since it is feasibly to increase the context awareness in the era of Internet of Things, it is promising to use the data collected by sensors to increase the accuracy of business process predictions. Fülop et al. [17] present in their paper a first conceptual framework for combining CEP with predictive analytics. Also Janiesch et al. [18] provide first conceptual works; however, they state that "[e]vent-based or event-driven business intelligence" approaches are only rudimentary researched and find limited integration in software tools. Even if Schwegmann et al. [19] already combines BAM with BPI, the question on how findings from this connection can be used for adapting and optimizing business processes is still open.

4.3 Business Process Adaptation and Optimization

In traditional business process management (BPM) approaches, processes are typically adapted and improved on type level at design time after passing a business process lifecycle. This ex post handling—as it is regularly applied in business process controlling and mining—however causes considerable time delay [20]. Since an aggregation of process execution data has to be done in a first place, before processes can eventually be optimized at type level, problematic process executions have already been completed and can no longer be optimized. Thus, scientists from the process mining community explicitly call for a stronger "operational support" [21]. This requires the existence of immediate feedback channels to the business process execution. Without the possibility of such feedback loops, only ex-post considerations

are possible. At this level of run time adaptation and optimization of business processes, first approaches already exist [18].

In the scope of commercial BPM systems, there are some early adopters characterized by the term "Intelligent Business Process Management Suites" [22]. For example, IBM's Business Process Manager provides means to analyze operations on instance level in real-time, to control them and to respond with further process steps in an ad-hoc manner [23]; yet, a fully automation cannot be attested. On the market of real-time CEP, further vendors are present; e.g. Bosch Software Innovations or Software AG have implemented several CEP functionality into their BPM solutions [22].

However, in order to speak of intelligent business process management systems, a more powerful automated analysis of the variety of events must be implemented as a basis for automated event-based predictions and process control [1].

5 Conclusion and Outlook

This paper outlined the potentials of event-based predictions in order to plan and eventually control business processes. Besides outlining these potentials, a general concept for event-based predictions had been conceived and the current state of the art was discussed. In addition, the article showed, by means of a case study, which process data a sample steel producing company can currently collect by its applied sensor technology. This data base is potentially available for event-based predictions of its manufacturing processes. Based on the description of data, which was collected and evaluated by the central department of information and communication technology of the company, the paper concluded that with current techniques and systems the available data cannot be analyzed in a reasonable time horizon to make sufficient business value out of it. Without dedicated big data analytics, the sample company will not be capable of exploiting the potential of its data. Hence, the paper should form a working and discussion basis for further research in big data analytics research.

Acknowledgments. This research was funded in part by the German Federal Ministry of Education and Research under grant number 01IS12050 (project IDENTIFY).

References

1. Krumeich, J., Weis, B., Werth, D., Loos, P.: Event-Driven Business Process Management: Where are we now? - A Comprehensive Synthesis and Analysis of Literature. Business Process Management Journal 20 (in press, 2014)
2. Lundberg, A.: Leverage Complex Event Processing to Improve Operational Performance. Business Intelligence Journal 11, 55–65 (2006)
3. Unni, K.: Steel Manufacturing Could Use More Sensing and Analysis (2012), http://www.sensorsmag.com/process-industries/steel-manufacturing-could-use-more-sensing-and-analysis-10249 (accessed March 5, 2014)
4. Pettey, C., Goasduff, L.: Gartner Says Between Now and Year-End 2014, Overlooked but Easily Detectable Business Process Defects Will Topple 10 Global 2000 Companies (2011), http://www.gartner.com/newsroom/id/1530114 (accessed March 5, 2014)

5. Dubé, L., Paré, G.: Rigor in Information Systems Positivist Case Research: Current Practice, Trends and Recommendations. MIS Quarterly 27, 597–636 (2003)
6. Benbasat, I., Goldstein, D.K., Mead, M.: The Case Research Strategy in Studies of Information Systems. MIS Quarterly 11, 369–386 (1987)
7. Kurbel, K.: Produktionsplanung und -steuerung im Enterprise Resource Planning und Supply Chain Management. Oldenbourg, München (2010)
8. von Ammon, R., Ertlmaier, T., Etzion, O., Kofman, A., Paulus, T.: Integrating Complex Events for Collaborating and Dynamically Changing Business Processes. In: Dan, A., Gittler, F., Toumani, F. (eds.) ICSOC/ServiceWave 2009. LNCS, vol. 6275, pp. 370–384. Springer, Heidelberg (2010)
9. Beyer, M.A., Laney, D.: The Importance of 'Big Data': A Definition (2012), http://www.gartner.com/doc/2057415 (accessed March 5, 2014)
10. Stäger, M., Lukowicz, P., Tröster, G.: Power and accuracy trade-offs in sound-based context recognition systems. Pervasive and Mobile Computing 3, 300–327 (2007)
11. Choi, J.: RFID Context-aware Systems. In: Turcu, C. (ed.) Sustainable Radio Frequency Identification Solutions, pp. 307–330. InTech, Rijeka (2010)
12. Wu, E., Diao, Y., Rizvi, S.: High-Performance Complex Event Processing over Streams. In: ACM SIGMOD 2006, pp. 407–418. ACM, New York (2006)
13. Wang, F., Liu, S., Liu, P., Bai, Y.: Bridging Physical and Virtual Worlds: Complex Event Processing for RFID Data Streams. In: Ioannidis, Y., et al. (eds.) EDBT 2006. LNCS, vol. 3896, pp. 588–607. Springer, Heidelberg (2006)
14. Taylor, K., Leidinger, L.: Ontology-Driven Complex Event Processing in Heterogeneous Sensor Networks. In: Antoniou, G., Grobelnik, M., Simperl, E., Parsia, B., Plexousakis, D., De Leenheer, P., Pan, J. (eds.) ESWC 2011, Part II. LNCS, vol. 6644, pp. 285–299. Springer, Heidelberg (2011)
15. Brito, A., Martin, A., Knauth, T., Creutz, S., Becker, D., Weigert, S., Fetzer, C.: Scalable and Low-Latency Data Processing with Stream MapReduce. In: CloudCom 2011, pp. 48–58. IEEE Computer Society, Washington (2011)
16. Linden, M., Felden, C., Chamoni, P.: Dimensions of Business Process Intelligence. In: Muehlen, M.Z., Su, J. (eds.) BPM 2010 Workshops. LNBIP, vol. 66, pp. 208–213. Springer, Heidelberg (2011)
17. Fülop, L.J., Beszédes, A., Tóth, G., Demeter, H., Vidács, L., Farkas, L.: Predictive complex event processing: a conceptual framework for combining complex event processing and predictive analytics. In: BCI 2012, pp. 26–31. ACM, New York (2012)
18. Janiesch, C., Matzner, M., Müller, O.: Beyond process monitoring: a proof-of-concept of event-driven business activity management. Business Process Management Journal 18, 625–643 (2012)
19. Schwegmann, B., Matzner, M., Janiesch, C.: A Method and Tool for Predictive Event-Driven Process Analytics. In: Alt, R., Franczk, B. (eds.) WI 2013, pp. 721–735. University of Leipzig, Leipzig (2013)
20. Loos, P., Balzert, S., Werth, D.: Controlling von Geschäftsprozessen. In: Jochem, R., Mertins, K., Knothe, T. (eds.) Prozessmanagement, pp. 443–471. Symposium Publishing GmbH, Düsseldorf (2010)
21. van der Aalst, W., et al.: Process Mining Manifesto. In: Daniel, F., Barkaoui, K., Dustdar, S. (eds.) BPM Workshops 2011, Part I. LNBIP, vol. 99, pp. 169–194. Springer, Heidelberg (2012)
22. Sinur, J., Schulte, W.R., Hill, J.B., Jones, T.: Magic Quadrant for Intelligent Business Process Management Suites (2012), http://www.gartner.com/doc/2179715/magic-quadrant-intelligent-business-process (accessed March 5, 2014)
23. IBM (eds.): IBM Business Process Manager Standard (2014), http://www-03.ibm.com/software/products/en/business-process-manager-standard (accessed March 5, 2014)

Big Data as Strategic Enabler - Insights from Central European Enterprises

Rainer Schmidt[1], Michael Möhring[2], Stefan Maier[2], Julia Pietsch[2], and Ralf-Christian Härting[2]

[1] Hochschule München, Lothstraße 64, 80335 München, Germany
[2] Hochschule Aalen, Beethovenstraße 1, 73430 Aalen, Germany
Rainer.Schmidt@hm.edu,
{Michael.Moehring,Stefan.Maier,
Julia.Pietsch,Ralf.Haerting}@htw-aalen.de

Abstract. Big Data increases the amount of data available for analysis by significantly increasing the volume, velocity and variety. Big Data is the coincidence of technological developments with a radical transformation of information flows. Beyond technological considerations only few, general analyses of the strategic impact of Big Data exist. Therefore, we designed a study analyzing the strategic impact factors of Big Data. The study is based on a survey from 148 responses of enterprise specialists and managers in the field of Big Data, originating from central European enterprises. Key findings are that idle data have a negative moderate effect and the type of business process has a positive effect on the perceived advantages of big data. Furthermore there is an effect of the perceived advantages of big data on the planned and current use.

Keywords: big data, strategy, business intelligence, empirical research, business information system.

1 Introduction

The continuous growth of data is a new challenge for enterprises and organizations. The volume of data surpassed 2.8 zettabytes in 2012 – a growth by a factor of 9 over five years [1] and is expected to continue its rise to a 50 times higher volume in 2020 [2]. Big Data is used for processing and analyzing large amounts of heterogeneously structured data with a high velocity. Therefore, Big Data [3] [4] has become a central topic in management [5]. The importance of Big Data for management is supported by an empirical study (worldwide online survey of over 1300 IT managers) from ZDNet which indicates that "70% will use data analytics by 2013" [6]. Big Data is regarded as a highly disruptive information technological development [7]. It has been declared a national challenge and priority by President Obama's administration [8]. Big Data is also a new frontier for business and social science [9]. Big Data is considered as positive factor for the firm performance [10]. Companies using Big

W. Abramowicz and A. Kokkinaki (Eds.): BIS 2014, LNBIP 176, pp. 50–60, 2014.

Data and Analytics outperform peers 5 percent in productivity and 6 percent in profitability [11]. Big Data helps companies to learn more about the alternatives customers look for when considering buying a product. In this way, important factors in the customers' purchasing decision can be identified. Furthermore, companies gain knowledge how customers compose their shopping baskets. In spite of these expectations associated with Big Data, there are also critical thoughts [12] [13]. E.g. the problems created by too much trust into data are discussed in [6].

Despite its huge impact on enterprises, research on the strategic impact of Big Data in detail is just beginning. An initial analysis of the transformational effect of Big Data on the IT-department is undertaken in [14], [15]. The influence of Big Data on the design of business processes is demonstrated with several examples in [4]. In [5] a Big Data adoption model is introduced. It differentiates three levels of analytics adoption: aspirational, experienced and transformed. The highest level (transformed) is achieved by enterprises using analytics not only to guide but to prescribe actions. One precondition for achieving the transformed layer is a strong ability to capture, aggregate and analyze data. However, there is no deeper analysis of the impact of Big Data on business processes. An initial classification of the impact of Big Data in different sectors is given in [16]. The benefits from the use of Big Data in different sectors are analyzed in [7] .

In summary, both a thorough analysis of Big Data and its enabling potential as well as a detailed analysis of the impact of Big Data on business processes are still missing. To close this gap, we designed a study that analyses the impact factors of the perceived advantages of Big Data, the effects of Big Data, the current and planned implementation of Big Data in different business processes and the use of Big Data for strategic initiatives. The study is based on a survey comprising 148 responses of enterprise specialists and managers in relation to Big Data, originating from central European enterprises. Having eliminated incomplete and inconsistent cases, 110 responses were used for data analysis. The online survey started in January 2013 and ended in May 2013.

Our paper proceeds as follows. First, we discuss the basic properties of Big Data. In particular, Big Data is distinguished from traditional business intelligence concepts. Furthermore, enterprise architecture patterns for Big Data in enterprises are identified. In chapter 3 we introduce the research design and create four hypotheses. In the following we present our research methods and data collection. The results of our research are given in chapter 4, we give an outlook and conclusion.

2 Big Data

Big Data is the coincidence of technological developments with a radical transformation of information flows within enterprises and organizations. Big Data is not a replacement for concepts such as Business Intelligence [17] and Business Analytics [18]. Instead it increases the set of data available for analysis by significantly increasing volume, velocity and variety of data processing [19]. Existing approaches are mostly constricted to the use of transactional data, e.g. describing the purchasing of

goods. These data are well-structured, often stored in relational databases, and follow clearly defined semantics. Big Data extends this data in several ways. First, non-transactional data, such as customer interactions logged in customer relationship management systems is integrated into analysis. Nevertheless, such data are still clearly defined in terms of semantics and structures. The next extension is the integration of web logs etc. from customer-interfacing information systems. These data are no longer well-structured but subject to frequent structural changes. However these data are machine-created which results in homogeneous semantics.

Big Data supplements existing business intelligence and business analytics approaches in three ways as shown in Fig. 1**.** Integration of Business Intelligence and Big Data [15]. The first way is to use Big Data as an extension to an existing business intelligence application. The Business Intelligence application consists of a data processing layer, an information processing layer and a presentation layer. The Big Data application provides information created from data that are too huge and/or too unstructured to be processed by the legacy business intelligence system. The second way for using Big Data is to present the information created by the Big Data application (with own data processing and information generation) using an own presentation layer. The third possibility is to support other applications such as web-shops (with standardized interfaces using data processing and information generation). E.g. Big Data provides product suggestions for increasing cross-and upselling. This approach can be used by different applications in the enterprise to generate cost savings by cutting down maintenance, support and total unit costs of transactions.

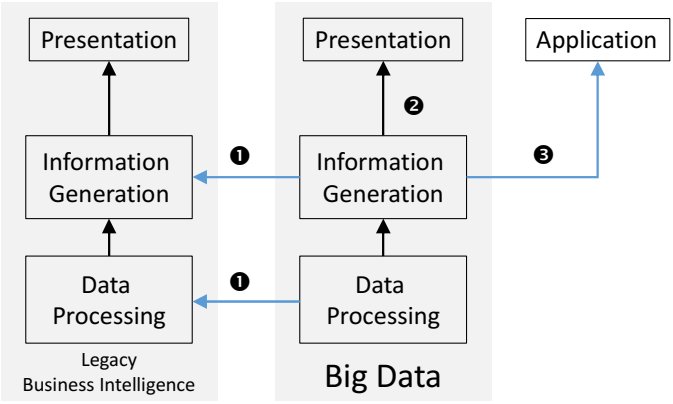


Fig. 1. Integration of Business Intelligence and Big Data [15]

3 Research Design and Method

In order to explore the potential of Big Data to enable a new performance of business processes and profitability we designed a study to identify key impact factors for implementing Big Data technologies and applications in business processes in enterprises and other organizations.

3.1 Design of the Study

The design of our study contains four hypotheses shown in Fig. 2. In companies and organizations enormous amounts of idle data exist [1]. Idle data are data available for analysis but not used today, due to restricted capacities and non-existent analytical capabilities of enterprise information technology. Therefore, the extent of existing idle data in a business process can be considered as an indicator for possible advantages by using Big Data.

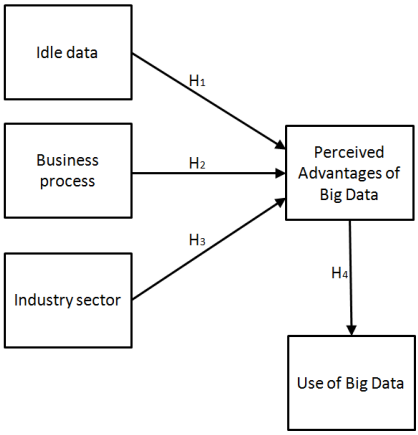


Fig. 2. Research Model

The first hypothesis of our study is, that the amount of idle data influences the perceived advantages of Big Data [19]. The perceived advantages of Big Data are defined as the capability of the use of Big Data in business processes (e.g. marketing) by data scientists and managers.

Hypothesis 1: The amount of idle data in business processes positively influences the perceived advantage of Big Data in business processes.

To study the impact of idle data, the online-based responses of the participants ranged on a scale of one to five (1: very low; 2: low; 3: moderate; 4: high; 5: very high) for business processes according to Value Chain introduced by Porter and Millar [3].

The advantages of Big Data may be different from business process to business process [3]. Literature often describe business cases of Big Data in marketing processes [4] [14]. Other core business processes (e.g. business processes of the human resources management) often ignored in publications. To investigate this topic, we created the second hypotheses, expressing that the type of business process has an influence on the perceived advantages of Big Data.

Hypothesis 2: The perceived advantages of Big Data are influenced by the respective business process.

The perceived advantages of Big Data ranged on a scale of one to five (1: very low; 2: low; 3: moderate; 4: high; 5: very high) for the respective business processes. The strategic effects of information technology vary strongly from industry sector to industry sector. Thus, we developed a hypothesis expressing that the perceived advantages of Big Data are dependent on the industry sector [20] [21].

The advantages of Big Data vary depending on business processes and industry sectors because of individual business cases. According to Gartner [22] investments in relation to Big Data vary from industry sectors. For instance retail and education enterprises more invested in Big Data as Manufacturing and Insurance enterprises [22]. To test the influence of industry sector, we created hypothesis 3:

Hypothesis 3: The industry sector exerts an effect on the perceived advantages of Big Data.

Our classification of industry sectors is based on the European Classification of Economic Activities (NACE) version 2 [23] .

Furthermore, it is interesting to understand how strong the perceived advantages of Big Data influence the current or planned use of Big Data in enterprises and organizations [24] because of the different current and planned investments related to Big Data [22]. So we defined the hypothesis 4:

Hypothesis 4: The perceived advantages of Big Data exert a positive effect on the current or planned use of Big Data.

The use of Big Data ranged from 1 to 4 as the following: 1: no plan of use; 2: use planned later than 2013; 3: use planned in 2013; 4: in use. The idle data, business process and the industry sector are important impact factors of the perceived advantages of Big Data and based on the current and planned implementation of Big Data in each business processes.

3.2 Research Methods and Data Collection

Regression analysis [25] [26] [27] is characterized as a family of methods for exploring and establishing a functional relationships between independent and dependent variables. The application of regression models is widespread use and applicable to many subject areas. Regression analysis can follow two basic approaches – simple regression and multiple regression. Generally, the regression analysis examines the relationship between a quantitative dependent variable and one or more quantitative independent variables (explanatory or predictor variables) [28].

To investigate the perceived and current or planned use of Big Data a local online-based survey in the German language (e.g. for enterprises of the countries Germany, Austria and Switzerland) were implemented. First a pre-test to reduce problems was executed, and findings were considered. The online survey started in January 2013

and ended in May 2013 (during five months). Hundreds of enterprise specialists in this area were invited to participate in this study. The final sample consists of n = 148 responses of data scientists and managers in relation to Big Data. After elimination of incomplete and inconsistent cases 110 responses were used for data analysis. To ensure that the participants were qualified to respond to this survey, we asked only specialists or managers from business or IT-departments with relations to Big Data and used validation questions (e.g. questions of their current job position within the enterprise).

The respondents cover a wide variety of industry sectors (based on European Classification of Economic Activities (NACE) version 2 [23] (Table 1). The majority of participating enterprises operate in the fields of transportation and storage, information and communications as well as manufacturing.

Table 1. Industry sectors of the participants enterprises

Industry sector	**Percent**
Transportation and storage, information and communication	37.27%
Manufacturing	20.91%
Activities of households as employers, undifferentiated goods- and services-producing activities of households for own use	10.00%
Real estate activities, professional, scientific and technical activities, administrative and support service activities	8.18%
Wholesale and retail trade, repair of motor vehicles and motorcycles	3.64%
Financial and insurance activities	3.64%
Education	3.64%
Arts, entertainment and recreation, other service activities	3.64%
Human health and social work activities	2.73%
Electricity, gas, steam and air conditioning supply, water supply, sewerage, waste management and remediation activities	1.82%
Construction	1.82%
Accommodation and food service activities	0.91%
Public administration and defence, compulsory social security	0.91%
Activities of extraterritorial organisations and bodies	0.91%
Agriculture, forestry and fishing	0.00%
Mining and quarrying	0.00%

Most respondents work for companies with more than 250 employees (30 percent work for companies with 500 or more employees).

4 Results

We used IBM Statistics SPSS 20 to test our hypotheses (Figure 2) employing linear regression models (H1, H3, H4) or descriptive analytics (H2). Hypotheses 1, 3 and 4 were verified by using bivariate linear regression modeling. The usage of this analysis consists of a dependent variable (e.g. perceived advantage of Big Data in each

business unit) and an independent variable (e.g. idle data in each business unit) in the same business process (e.g. marketing). The variable-based impact was analyzed by non-standardized and standardized (beta) coefficient of regression as well as the significance (p-value).

To discover how the idle data impacts the perceived advantages of Big Data in each business process a linear regression with the "perceived advantages of Big Data" as the dependent variable and the "idle data" as the independent variable in each business process is applied (Table 2).

Table 2. Linear regression for hypothesis 1

Variable	Regr. coeff.	Beta	Sign.
Inbound logistics	0.268	-0.462	0.000
Outbound logistics	-1.388	-0.361	0.000
Operations	-1.486	-0.438	0.000
Marketing	-1.299	-0.397	0.000
Management	-1.125	-0.329	0.002
Financial Service	-1.026	-0.265	0.015
Technology development / IT	-1.093	-0.346	0.001
Human Resource Management	-0.939	-0.333	0.002

The comparison of the beta values (effect of the independent variable on the dependent variable) of each business process shows a moderate negative and significant effect of the availability of idle data to the perceived advantage of Big Data in each business process (from -0.265 to – 0.462). Therefore, hypothesis 1 (The amount of idle data in business processes positively influences the perceived advantage of Big Data in business processes.) must be rejected.

Table 3. Descriptive Analytics for Hypothesis 2

Variable	Average
Inbound logistics	3.018
Outbound logistics	2.736
Operations	3.109
Marketing	3.409
Management	3.200
Financial Service	2.727
Technology development / IT	3.572
Human Resource Management	2.618

Furthermore, the study shows differences in the perceived advantage of Big Data for every business process. Table 3 shows the averages for the respective business processes. The perceived use of Big Data related to each business process is rated on a five-grade scale (1: very low; 2: low; 3: moderate; 4: high; 5: very high):

Technology development / IT (average: 3.572) and Marketing (average: 3.409) are the core processes with the highest perceived use of Big Data (see Table 3).

Especially in the marketing environment applications of Big Data enable new knowledge transfer to the management. That helps to get a better understanding of important marketing issues (e.g. forecast the success of a marketing campaign).

To answer these questions plenty of applications are available. Based on a Big Data fundament tools like decision trees, association, time series, clustering or neural nets are often used.

In contrast to the Marketing the processes Human Resource Management (average 2.618) and Outbound Logistics have the lowest use perceived. All core business processes have a noticeable significant influence of the perceived use of Big Data (highly two-side significant (0,000) of 1 and 2 (very low and low perceived advantages of Big Data)). Therefore, hypothesis 2 (The perceived advantages of Big Data are influenced by the respective business process) is supported by the survey data. Hypothesis 3 tests the influence of industry sector. The data collection for hypothesis 3 was focused on the usage of Big Data in different business sectors. Therefore, the identification of 8 business sectors was made.

Table 4. Linear regression for hypothesis 3

Variable	**Regr. coeff.**	**Beta**	**Sign.**
Inbound Logistics	0.209	0.099	0.397
Outbound Logistics	0.423	0.182	0.118
Operations	0.894	0.353	0.002
Marketing	-0.207	-0.084	0.472
Management	0.135	0.058	0.622
Financial Service	0.097	0.059	0.613
Technology development / IT	-0.256	-0.138	0.239
Human Resource Management	-0.010	-0.009	0.939

After analyzing the received data, it was obvious that - based on the measure of determination of the significance - the results could not be interpreted well. Consequently a reduction and selection of the three dominating business sectors within the sample was made to improve the chance for significant results:

- Transportation and storage, information and communication (n=41)
- Manufacturing sector (n=23)
- Activities of households as employers; undifferentiated goods- and services-producing activities of households for own use (n=11)

After that, the significance levels improved but remained mostly unsatisfactory. The only significant business process within this model is operations. That means that there exists a relation between the investigated business sector and the perceived advantage of the business process. Nevertheless if the attention is paid to the coefficient of regression and the Beta coefficient the data can be interpreted as follows: The coefficient of regression is between -0.256 and 0.894 what results in an arithmetic average of 0.160625. The Beta coefficient contains an averaged value of 0.065 (the single data range between -0.138 and 0.353). That means that in comparison of both standardized

and non-standardized values a significant impact is not visible. Ergo hypothesis 3 (The industry sector exerts an effect on the perceived advantages of Big Data.) cannot be confirmed - that is to say that the business sector has no impact on the perceived advantages of Big Data. But such a conclusion would be based on non-significant values.

The effect of the perceived advantage of Big Data on the planned and current use shows the following results based on a linear regression model (dependent variable: the planned and current use of Big Data; independent variable: perceived advantages of Big Data):

Table 5. Linear regression for hypothesis 4

Variable	Coeff. of regr.	Beta	Sign.
Inbound logistics	0.243	0.332	0.004
Outbound logistics	0.181	0.330	0.004
Operations	0.179	0.299	0.009
Marketing	0.402	0.537	0.000
Management	0.251	0.385	0.001
Financial Service	0.327	0.383	0.001
Technology development / IT	0.387	0.494	0.000
Human Resource Management	0.493	0.310	0.007

On average, there is a low moderate effect (average of beta value: 0.383) on the current and planned use of Big Data based on the perceived advantages of Big Data. The maximum value of beta is 0.537 (Marketing), the minimum value is 0.299 (Operations). Therefore, hypothesis 4 (The perceived advantages of Big Data exert a positive effect on the current or planned use of Big Data.) can be confirmed.

5 Conclusion

Research on the strategic impact of Big Data in detail is just taking off. Our research shows that Big Data has a high perceived advantage in many business processes and especially in Technology development and IT as well as in Marketing. Thus, enterprises can define their investments in Big Data technology related to these primary processes and their business needs. Especially small and medium sized enterprises are able to focus their Big Data initiatives on investments providing the highest business impact and can so reduce financial risks.

Furthermore, we found that the business sector shows no significant influence on the perceived advantages of Big Data. Future research based on larger samples may yield different findings. The idle data in each business process has a negative effect on perceived use of Big Data and the perceived advantages of Big Data influence its current and planned usage.

This research has both practical and theoretical implications. Managers can benefit from information about the use of Big Data in different business processes or units, especially in Marketing and Technology development / IT. Academic research can

benefit of new knowledge about differences in the use of data analysis in different core processes and sectors of central European enterprises. Consequently methods and theories of data analysis and Big Data / Business Intelligence can adopt these results in order to create or improve current approaches (e.g. for implementing management information systems).

Generally the study results are significant. But there are exceptional cases. So there is a need for future research and a comparison in few years later to document the effect of the use of Big Data after several implementation projects. In our study we focused on the most abstract business processes such as operations, marketing, etc.. In the future there is a need to explore business processes in detail (e.g. process of customer relationship management or sales) and to evaluate the best implementation points for Big Data. That helps to improve business processes (e.g. reduce process time and process costs, improve process quality) in detail. In addition a framework for implementation of Big Data can be built. Not in all business cases better decisions based on a larger data base can be implemented. Therefore a proof of concept for each business case of Big Data is needed. Future research will enlarge the number of countries involved. For example enterprises from North America and BRIC states are very interesting, because of their economic growth and standing. A comparison of the Big Data adoption in these countries with the one in central Europe, will be an interesting theme for research.

References

1. Gantz, J., Reinsel, D.: Extracting value from chaos. IDC IView, 1–12 (2011)
2. Mearian, L.: World's data will grow by 50X in next decade, IDC study predicts, http://www.computerworld.com/s/article/9217988/World_s_data_will_grow_by_50X_in_next_decade_IDC_study_predicts
3. Porter, M.E., Millar, V.E.: How information gives you competitive advantage. Harv. Bus. Rev. 63, 149–160 (1985)
4. McAfee, A., Brynjolfsson, E.: Big data: the management revolution. Harv. Bus. Rev. 90, 60 (2012)
5. LaValle, S., Lesser, E., Shockley, R., Hopkins, M.S., Kruschwitz, N.: Big data, analytics and the path from insights to value. MIT Sloan Manag. Rev. 52, 21–32 (2011)
6. Bursting the Big Data Bubble | ZDNet, http://www.zdnet.com/bursting-the-big-data-bubble-7000002352/
7. Bughin, J., Chui, M., Manyika, J.: Clouds, big data, and smart assets: Ten tech-enabled business trends to watch. McKinsey Q. 56 (2010)
8. Weiss, R.: OBAMA ADMINISTRATION UNVEILS "BIG DATA" INITIATIVE
9. Kaisler, S., Armour, F., Espinosa, J.A., Money, W.: Big Data: Issues and Challenges Moving Forward. In: 2013 46th Hawaii International Conference on System Sciences, pp. 995–1004. IEEE Computer Society, Los Alamitos (2013)
10. Brynjolfsson, E., Hitt, L., Kim, H.: Strength in Numbers: How Does Data-Driven Decisionmaking Affect Firm Performance? (2011)
11. Breuer, P., Forina, L., Moulton, J.: Beyond the hype: Capturing value from Big Data and advanced analytics, http://cmsoforum.mckinsey.com/article/beyond-the-hype-capturing-value-from-big-data-and-advanced-analytics

12. Gray, J.: What data can and cannot do, http://www.guardian.co.uk/news/datablog/2012/may/31/data-journalism-focused-critical
13. The Hidden Biases in Big Data - Kate Crawford - Harvard Business Review, http://blogs.hbr.org/cs/2013/04/the_hidden_biases_in_big_data.html
14. Möhring, M., Schmidt, R., Wolfrum, N., Kammerer, M., Maier, S., Höritz, S.: How Big Data Transforms the IT Department to a Strategic Weapon. In: Proceedings of the IADIS International Conference Information Systems 2013, Lisbon, pp. 323–326 (2013)
15. Schmidt, R., Möhring, M.: Strategic alignment of Cloud-based Architectures for Big Data. In: Proceedings of the 17th IEEE International Enterprise Distributed Object Computing Conference Workshops (EDOCW), Vancouver, Canada, pp. 136–143 (2013)
16. Manyika, J., Chui, M., Brown, B., Bughin, J., Dobbs, R., Roxburgh, C., Byers, A.H.: Big data: The next frontier for innovation, competition, and productivity. McKinsey Glob. Inst., 1–137 (2011)
17. Kemper, H.-G., Baars, H., Lasi, H.: An Integrated Business Intelligence Framework. In: Rausch, P., Sheta, A.F., Ayesh, A. (eds.) Business Intelligence and Performance Management, pp. 13–26. Springer, London (2013)
18. Barton, D.: Making Advanced Analytics Work For You. Harv. Bus. Rev. 90, 78–83 (2012)
19. Chen, H., Chiang, R.H.L., Storey, V.C.: Business intelligence and analytics: From big data to big impact. MIS Q. 36, 1165–1188 (2012)
20. Brown, B., Chui, M., Manyika, J.: Are you ready for the era of "big data"? McKinsey Q. 4, 24–35 (2011)
21. Hannula, M., Pirttimäki, V.: Business intelligence empirical study on the top 50 Finnish companies. J. Am. Acad. Bus. 2, 593–599 (2003)
22. Kart, L., Heudecker, N., Buytendijk, F.: Survey analysis: big data adoption in 2013 Shows Substance behind the hype (2013)
23. Commission, E.: NACE Rev. 2–Statistical classification of economic activities in the European Community. Luxemb. Off. Off. Publ. Eur. Communities (2008)
24. Mayer-Schönberger, V., Cukier, K.: Big data: a revolution that will transform how we live, work and think. John Murray, London (2013)
25. Stigler, S.M.: The Story of Statistics: The Measurement of Uncertainty Before 1900. Harvard University Press (1986)
26. Boscovich, R.J.: De litteraria expeditione per pontificiam ditionem, et synopsis amplioris operis, ac habentur plura ejus ex exemplaria etiam sensorum impressa. Bononiensi Sci. Artum Inst. Atque Acad. Comment. 4, 353–396 (1757)
27. Boscovich, R.J.: De recentissimis graduum dimensionibus, et figura, ac magnitudine terrae inde derivanda. Philos. Recentioris Benedicto Stay Romano Arch. Publico Eloquentare Profr. Versibus Traditae Libri X Adnot. Suppl. P Rogerii Joseph Boscovich SJ Tomus II, 406–426 (1757)
28. Mason, C.H., Perreault Jr., W.D.: Collinearity, power, and interpretation of multiple regression analysis. J. Mark. Res., 268–280 (1991)

App'ification of Enterprise Software: A Multiple-Case Study of Big Data Business Applications

Stefan Wenzel[1,2]

[1] Otto-Friedrich-Universität Bamberg, An der Weberei 5, 96047 Bamberg, Germany
[2] SAP AG, Dietmar-Hopp-Alle 16, 69190 Walldorf, Germany
stefan.wenzel@sap.com

Abstract. The success of consumer app stores is driving enterprise software vendors to establish their own online software sales channels. The enterprise applications need to adapt to this new go-to-market model so that business users can evaluate and acquire these solutions online via self-service. Big data applications are among the latest IT innovations in the enterprise segment. Therefore, this study investigates how these new applications are marketed and distributed using enterprise app stores. A multiple-case study and an app'ification assessment model are used to produce insights into this phenomenon and to identify tangible measures to increase the app store readiness.

Keywords: App Store, Enterprise Application Software, Business Applications, App'ification, Big Data.

1 Introduction

Consumer app stores in the mobile software segment have been very successful and providers never miss an opportunity to report new record numbers: At the end of 2012 Apple alone reported approximately 40 billion app downloads from their iTunes store [1]. The broad adoption of consumer technologies has also changed how business users rate corporate information technology (IT) and information systems (IS). The trend of technologies emerging in consumer markets and diffusing into the corporate segment is also referred to as "Consumerization of IT" [2]. Moreover, business end users are nowadays much more knowledgeable and sensitive towards the technology they use and either want to choose the software and IT themselves, or ask for more involvement in the decision and selection process [3]. As a result, business departments are gaining more influence in IT purchase decisions and the roles of IT departments and CIOs are being questioned [4, 5]. The rise of cloud computing in this context has only intensified this discussion [6, 7].

As a consequence, enterprise software vendors who traditionally market their software via personal sales channels to CIOs [8] are establishing new go-to-market models by trying to reproduce the success of consumer app stores: They are establishing their own online software distribution channels[1] and equipping their

[1] E.g., www.sapstore.com, pinpoint.microsoft.com, appexchange.salesforce.com

W. Abramowicz and A. Kokkinaki (Eds.): BIS 2014, LNBIP 176, pp. 61–72, 2014.

applications with consumer-oriented user interfaces [9] to address individual business users or business departments directly. However, the adoption rates and the maturity of the related e-commerce process of the enterprise app stores are still low [10]. Novelli and Wenzel argue that enterprise application software (hereafter abbreviated to enterprise software) is often not ready to be sold online – applications are still too complex and have a monolithic structure [11]. Hence, the acquisition of such software requires a consultative sales approach and conflicts with the rather "transactional" nature of app stores. Their reasoning is largely applicable to traditional enterprise applications (e.g., ERP, CRM), which often have a history of 10-20 years and a go-to-market model tailored to fit a consultative, direct, "offline" sales channel.

New IT innovations should therefore recognize new go-to-market models and adopt characteristics which enable online sales and business-driven adoption. Big data and associated analytics are among the major technology trends: Both analytics and big data were named by Gartner as being among the top ten strategic technologies for 2012 & 2013 [12, 13], and are ranked as key issues for IT executives - with business intelligence (BI) ranked first and big data ranked tenth [14]. Hence, vendors are releasing new analytical applications connected to big data repositories which are often deployed in the cloud with user interfaces optimized for mobile devices [12].

The research objective of this study is to evaluate this new breed of big data business applications and their readiness for the app store model. The app store model is defined as an online marketing, sales and distribution channel for software products or services [11]. Furthermore, the app store model for enterprises comes with a change in the enterprise software adoption paradigm: from a top-down, IT-governed model, to a bottom-up, business-driven approach [3]. The metaphor App'ification is used for applying a set of characteristics associated with (consumer) mobile apps to a different context – in this study the context is big data business applications [11].

A qualitative research strategy has been chosen and a case study design has been used. Multiple business applications have been studied within a single organizational context and a focus on idiographic and comparative analysis [15]. The individual cases (i.e., the applications) are evaluated using a pre-defined evaluation scheme and an "app'ification assessment model" derived from the literature. The actual assessment and other interpretative tasks in the case studies have been performed together with experts in the domain of enterprise software. SAP, one of the largest business software application vendors, was chosen to provide the organizational context since it fulfilled three important criteria: SAP has established its own version of an enterprise app store [16], they have released an application platform and infrastructure for the development and operation of big data applications (i.e., SAP HANA), and they have released dedicated big data business applications [17].

The in-depth analysis of the cases in this study reveals a number of different measures to increase the readiness for the app store model. Typical barriers for enterprise software and issues inherent for big data applications are discussed, and potential solutions are provided. The results can be used by IS researchers and practitioners. IS research (ISR) will benefit from the insights generated with regard to the under-investigated "app phenomenon", i.e., application characteristics that are relevant for online sales and distribution. Further, IS researchers can use the proposed app'ification assessment model and analysis framework for future qualitative or quantitative research projects. Business software vendors or IT departments can use

the findings to app'ify their software solutions or to evaluate ongoing app store initiatives.

2 Related Work

This study touches multiple fields of research in IS (technology acceptance, adoption and diffusion, e-commerce, software design, consumerization of IT, app stores), but also in marketing (organizational and individual buying behavior, software procurement). In this section, the focus is on those theories and works that have been used to perform the research or those supporting its relevance.

Consumerization of IT describes the emergence of technologies in consumer markets and their subsequent diffusion or even uncontrolled infiltration into business segments. To support the evidence of this trend, Harris et al. surveyed 4017 employees [18]: 52% responded that they would at least sometimes use their personal consumer device for work related activities, 36% stated that they would not worry about IT policies and just use the technology they need to perform their work, and 45% agreed to the statement that private devices and software applications are more useful than the ones provided by corporate IT. Weiß and Leimeister describe the opportunities as well as the risks of this trend for corporate IT, including IT governance and management of IS [2]. However, the uncontrolled infusion of such technologies threatens typical IT targets ("shadow IT", [19]), such as security, integrity, or integration. Beimborn et al. expect to effectively counter the issues arising with shadow IT and meet employees' expectations if enterprise software would be more consumer-like and if business-oriented distribution channels such as enterprise app stores could be used [20].

Novelli & Wenzel have therefore investigated the use of an **online sales channel** (i.e., enterprise app store) for the acquisition of enterprise software [11, 21]. They identified drivers and barriers related to using an online channel from an organizational point of view. The drivers and barriers were grouped into three categories: solution attributes, transactional attributes, and customer attributes. Solution attributes describe characteristics of enterprise software applications (e.g., ERP solution, ERP add-on, mobile business solution) that have an influence on the adoption of the online channel. Transactional attributes describe circumstances unique to an individual buying situation; customer attributes describe characteristics of the acquiring organization [11]. Solution attributes are the ones with the largest impact on the adoption of the online channel, as they not only influence the adoption directly, but they also have a major impact on the other attributes, but not vice versa. Table 1 shows the solution attributes and the corresponding "app'ified" instances. The solution attributes have been used as a basis to assess the app'ification of the investigated software solutions. However, they could not be used as they stand, since they evaluated the adoption of an online channel from an organizational point of view. Therefore, some criteria were grouped together and others had to be interpreted from a business user's or department's point of view. The following assessment criteria have been derived from Novelli & Wenzel's work [11]: "Task-oriented scope" (scope), "trialability" (evaluability), "starter package" (low price level), "instant use"

Table 1. App'ification criteria according to Novelli & Wenzel [11]

Solution attributes influencing online purchase of Enterprise SW	Ideal type of app'ified Enterprise SW
Scope	Focused / task-oriented scope
Implementation	Instant usage
On-demand delivery	Cloud infrastructure
Number of users	Low number of users
Evaluability	Trial available
Price level	Low entry price / starter package available
Customization	No customization
Criticality	Low criticality

(implementation, customization), "minimum infrastructure requirements" (on-demand delivery), "business-user-driven adoption" (criticality, number of users).

Because the assessment in this study does not only evaluate the eligibility of solutions for online sales, but also if they are suited to be adopted by business users or departments, the second theory that has been used to develop the assessment model is the **Technology Acceptance Model** (TAM). It is one of the most frequently used theories for technology acceptance in ISR and was first presented by Davis in 1989 [22]. In its original version, the theory states that two variables mainly impact a user's attitude towards using a technology - perceived usefulness and perceived ease of use. Multiple extensions of the TAM have been introduced since then and have enhanced the two utilitarian variables, e.g., with hedonistic concepts such as "perceived enjoyment", "objective usability", "enjoyment", or "design aesthetics". These can be seen as either additional and independent variables towards the intention to use, or as determinants of "perceived ease of use" [23, 24]. In this study, these individual aspects of technology acceptance have been considered by using the assessment criterion business-user-driven adoption". This criterion was not used to actually rate variables such as "ease of use", "design", or "perceived usefulness", but only to assess whether a business user would be able to evaluate these variables for a given business application in a self-service.

Verville and Halingten presented a **process model for the purchase of ERP** software [8]. They proposed a six-stage model including planning, information search, selection, evaluation, negotiation, and choice. This process model was used as a basis for the understanding of the buying process of software applications and to assess the "completeness of [a solution's] e-commerce process".

3 Methodology and Research Process

A qualitative research strategy using a multiple-case study design has been chosen to address the research objectives. The focus is on ideographic and comparative results. Case study design in general is used to investigate the unique characteristics of the selected case in its particular context (i.e., ideographic). Multiple case studies also allow comparisons (i.e., comparative case study) among the selected cases and are a

variant of theoretical replication [15, 25]. The unit of analysis is the individual business software application in the domain of analytical, big data applications. Publicly available documents and information have been reviewed and analyzed. The research process was conducted in seven sequential steps (cf. Fig. 1).

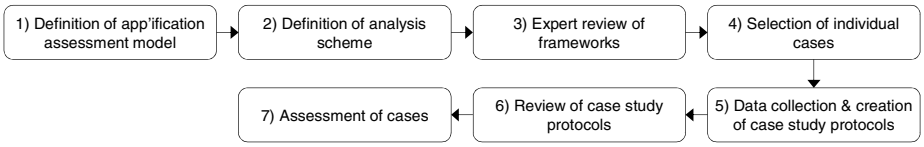


Fig. 1. Simplified sequential representation of the research process

Step 1. First an app'ification assessment model was derived based on the previously presented literature (cf. Fig. 2). **Step 2.** A standardized analysis scheme was created (cf. Fig. 2). The analysis scheme ensured that the multiple case studies could be replicated and that the case study data was captured as completely as possible. This approach ensured that the information gathered could be compared, since different data sources, level of information, and terminology were unified. Additionally, this method supports the objectivity of the research by avoiding "cherry picking" during data collection. The analysis scheme was defined by the example of the business model definition by Stähler [26] and adopted to this study's context. It consists of three views (commercial, functional, and technical view) and sub-categories. The sub-categories in each view have been defined by executing an explorative test case study and analyzing the data (cf. Table 2).

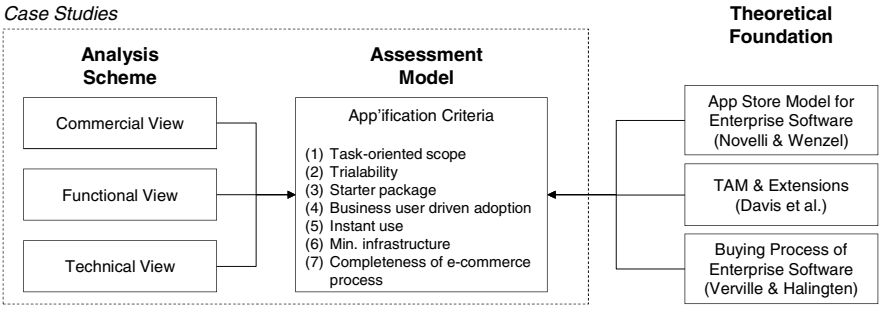


Fig. 2. Frameworks used to conduct the case studies

Step 3. Both the assessment model and the analysis scheme have been presented to and discussed with an expert group, which also supported the analysis of the individual cases. The expert group consisted of five experts with the following functional job roles: software architect (1), sales manager (1), e-commerce specialist (1), product manager (2). All experts are employed in the enterprise software industry and have at least six years of experience in their role.

Table 2. Case study analysis scheme: Views and sub-categories

Analysis Views	Sub-Categories
Commercial View	▪ Pricing and Licensing ▪ Packaging and Upsell Model ▪ Online Marketing / Online Sales ▪ Sales Target Level / Level of Adoption ▪ Online Evaluability / Trialability ▪ Pricing and Licensing
Functional View	▪ Claimed Value Proposition ▪ Functional Category and Industry Focus ▪ Visual and Interaction Design ▪ Scope Design ▪ Business Criticality
Technical View	▪ Architecture and Technological Foundation ▪ Device Support ▪ Implementation and Deployment ▪ Integration and Governance

Step 4. SAP was chosen to provide the organizational case context, since it is one of the major business application providers, has established online software sales channels [16, 27], a dedicated big data and analytics platform (SAP HANA), and has already released standard business applications in the area of big data [17]. Furthermore, SAP provides a program for independent software vendors to develop their own solutions on top of SAP HANA and market these solutions via the SAP Store [28]. The selection of the individual cases was done in two steps: Firstly, a shortlist was created based on the following hard criteria: Applications are published on the SAP Store [16] and/or on the SAP HANA Marketplace [27], the applications qualify as big data applications or real-time analytical or high-performance applications, and they are not merely infrastructure products, but business applications. The ca. 1400 available solutions on both online stores were reduced to a shortlist of 36 solutions. Subsequently, these 36 solutions were screened taking into consideration the following soft criteria: Applications are typical instances for their type of software, applications represent the breadth of big data applications available, and sufficient public material is available online, thereby allowing in-depth analysis of the defined analysis categories. Ultimately, five solutions were selected for the multiple case study (cf. Table 3). Four solutions are directly provided by SAP and one solution is provided by the SAP partner Liquid Analytics.

Step 5. The data has been sourced using publicly available documents, and has been collected using the following online sources: SAP Homepage (general product information, [29]); SAP Store ("SAP's App Store", [16]); SAP HANA Portal (dedicated portal on SAP's in-memory database and application platform for big data applications, [30]); SAP HANA Marketplace (department of SAP Store for solutions based on SAP HANA, [27]); SAP Community Network (blogs, wikis, forums, [31]); SAP Help Portal (product documentation, [32]). For the one non-SAP solution, the homepage of Liquid Analytics was used in addition [33]. The data was recorded and

prepared in case protocols (56 pages in total, 8-11 pages per case), which are structured according to the analysis scheme (cf. Table 2).

Step 6. Subsequently, two consecutive workshops (~120 minutes each) were conducted with the aforementioned expert group and with one researcher as the moderator. The first workshop focused on reviewing the case study protocols and discussing those parts that included subjective or interpretative information that resulted from the need to aggregate and categorize the information.

Step 7. The assessment of the individual cases was performed in a second workshop using the assessment model and the case study protocols. Deviating opinions were discussed until a consensus was reached. The results of the discussions were directly transcribed during the workshops.

4 Presentation of Cases

In this section, the cases and the results of the app'ification assessment are presented. A general overview is given, and subsequently each of the five cases is presented with the focus on selected unique and representative findings.

Table 3 gives an overview of the app'ification assessment. The degree of fulfillment for each of the criteria was rated on a five-level scale (very low, low, medium, high, and very high). If no assessment was possible due to lack of sufficient public information, the criterion was marked with "not applicable" (N/A). For every investigated criterion, there was at least one case with a "very high" rating. However, none of the researched solutions was rated high or very high in every aspect.

Table 3. Overview app'ification assessment

Name / Criteria	SAP Lumira	SAP Sales & Operations Planning	SAP Customer Value Int.	SAP Account Intelligence	Decisions by Liquid Analytics
Task-oriented scope	Medium	High	Medium	Very High	Very High
Trialability	Very High	High	Low	Very High	High
Starter package	Very High	N/A	Medium	High	Medium
Minimal infrastructure	Very High	High	Low	Low	Low
Instant use	Very High	Medium	Very Low	Medium	Low
Busines- user-driven adoption	Very High	Low	Low	Medium	Medium
E-Commerce Process	Very High	Low	Low	Very High	High

SAP Lumira. SAP Lumira is a self-service BI-platform to prepare and visualize data from corporate, personal, and ad-hoc data sources of any size (from small data sets to big data). It is available as a desktop edition and a cloud edition. Its strengths lie in data visualization, combined with an easy-to-use user interface. The tool targets data analysts or business analysts, from occasional to power users.

Since SAP Lumira is an entire productivity suite for various data analysis tasks, ranging from data preparation & cleansing, and data enrichment to data visualization and presentation, its task-orientation has been rated as medium. Trialability of the solution is very high: There are free versions available for the desktop and the cloud versions. Both free versions provide a feature set that is capable of conducting real-world tasks. Furthermore, a full feature version without limitations can be tested for 30 days. Multiple video and text tutorials help users to get started in a self-service.

The application can be used as a standalone version and does not need to be integrated to be used. Integration into corporate databases works via available database drivers. If the database already provides services and open ports (in the intranet) the connection to a database can be performed by the user in a self-service. Hence, the application meets the criterion of having minimal infrastructure prerequisites to a large extent. The business-user-driven adoption criterion is largely fulfilled, since the application explicitly does not require IT involvement for productive use and any business user can evaluate its usefulness and ease-of-use in a self-service. The related e-commerce process is also very complete in that it can be found, evaluated, directly purchased, downloaded, and activated online.

SAP Sales and Operations Planning (S&OP). S&OP is a decision support system and planning solution to support the entire process of matching demand and supply. S&OP uses a unified model including demand, supply, and financial data, and is organized as a workflow. It is based on SAP HANA for interactive, real-time, what-if analyses, and is available either in the cloud or on-premise.

S&OP supports the entire demand and supply matching process and therefore involves different roles (e.g., planning analysts, sales and marketing managers, finance experts) and departments in the company. Because it can be used in a restricted environment to only support the core demand and supply matching task, its task-orientation is rated as high. Trialability is also rated as high, since a free 3-day trial with sample data is available with most functionality enabled. However a full proof-of-concept will not be possible due to the limited time and functionality of the trial and the complexity of the solution. S&OP is available as a native cloud version and does not require additional infrastructure in this edition. For integration with other corporate systems, either the available middleware or a SAP HANA cloud integration service can be used. The minimal infrastructure criterion is therefore rated as high. However, the configuration of the S&OP data model and integration of the solution with corporate systems requires a dedicated implementation project. This process is well documented and standardized; “instant use” can therefore be rated as medium.

Business-user-driven adoption and completeness of the e-commerce process have both been rated as low, since the solution does not target a single role and can only fully be evaluated and acquired in cooperation with multiple organizational stakeholders. The further setup and integration of the product requires significant involvement of the IT department. The sales process of the solution is only partly reflected on the available online channels. Trial and product information are available, but a sales representative has to be contacted to actually purchase the product.

SAP Customer Value Intelligence (CVI). CVI analyzes the present and future value of customers. It provides a single view of customer value to target specific segments, offers selling recommendations, and supports sales and marketing departments with

investment decisions. It is designed as an interactive, real-time analytical solution capable of processing a large amount of transactional data from multiple sources. It can be used for everything from strategic customer segmentation to customer individual targeting and whitespace analysis.

CVI is a typical departmental solution (i.e., for sales and marketing departments) and supports the complex process of customer segmentation and targeting on multiple levels by providing an extensive set of tools. The task orientation criterion is therefore only partially fulfilled . To evaluate the solution only a "clickable" demo is available to highlight key functionality along a pre-defined path. To start off, a so-called "Rapid Deployment Solution" (RDS) is available with a predefined scope and implementation activities for initial and incremental adoption of the solution. However, a single starter package with fixed pricing at a low level is not offered.

From an infrastructure perspective, a dedicated SAP HANA instance needs to be acquired first, which is also available as a hosted variant. Further, the solution relies on specific backend systems (SAP ERP or CRM) and a data replication infrastructure. This, in turn, requires significant implementation effort before the solution can be used. Business-user-driven adoption is therefore also at a low level. Single users or even departments are not enabled to completely evaluate and decide upon the adoption on their own; IT departments are required for fully integrating the solution before it can be used productively. From an e-commerce perspective, product information is available and the evaluation phase can partly be conducted online. For deeper evaluation and purchasing, a sales representative has to be contacted.

SAP Account Intelligence (AI). AI is an enhancement to the previously presented solution CVI and comes as an app for the iPad. It provides an overview of all customers and helps sales representatives to target and analyze customer accounts in real time using the data from the underlying CVI solution. It is designed around a user-friendly user interface and makes extensive use of geographical data and map views to analyze key sales-oriented performance indicators.

The task-orientation of the solution is rated as very high, since it is tailored to sales representatives and is used by them to better plan customer engagements. Trialability is also possible: The app can be downloaded for free and includes a fully functional demo mode with sample data. AI does not come as a standalone application, but as an extension to CVI. Assuming that CVI is in place, the entry barriers are rather low. A single user-based fixed price allows an incremental adoption of the solution (i.e., starter package criterion). From an infrastructure perspective, the CVI solution and a gateway solution need to be in place. If this is the case, only very few activities need to be performed to instantly use the application. Business users are in a position to properly evaluate the solution and make an adoption decision on their own. They even might acquire the solution without further IT involvement if the CVI solution is already in place. Also the e-commerce process is very complete – information, evaluation, purchase, and distribution can be conducted online.

Decisions by Liquid Analytics. Decisions is both an analytical and a transactional solution for travelling salesmen in the area of wholesale, direct store distribution, and warehousing. It uses predictive analytics to provide real-time restock, cross- and up-sell, and goal-related product recommendations. It enables sales persons to manage the utilization of retailers' shelf space, recommend cross-sell and up-sell products,

and meet their sales goals. Furthermore, orders can be booked and verified in the application. It is available on mobile tablets and is based on SAP HANA.

Task-orientation is very high, since it is designed specifically for the travelling sales person and his day-to-day tasks for planning and conducting customer meetings. A trial version is also available as part of the freely downloadable app. The users have to pay for the cloud-based backend application, which comes as a monthly subscription per user. This allows companies to start small, with only a few subscribed users and to add more users incrementally. However, a required implementation will necessitate a certain upfront investment – therefore, the starter package availability is rated as medium. Business users will find it easy to evaluate usefulness, ease-of-use, and visual experience on their own. The application will need to integrate with business systems, such as ERP or CRM systems for productive use. The required SAP HANA instance and SAP Mobile Platform can be deployed in the cloud. Overall integration and infrastructure requirements are quite significant. Instant use of the application is therefore not possible. The e-commerce process is relatively complete – it is offered on the SAP Store, a trial is available online, and it can be purchased directly. However, detailed technical information is not available online.

5 Interpretation and Discussion of Results

The presented solutions are very heterogeneous with regard to scope, application domain, complexity, and technology. By studying the cases, different instruments were identified to fulfill the app'ification criteria.

Task-oriented scope is provided if a solution is designed for a very specific task or a specific business role (e.g., Decisions for travelling sales people). But also applications covering entire departments or cross-departmental processes can increase their task-orientation if they allow partial use (S&OP) or if certain functionality is excluded and offered as an enhancement (AI). Task-orientation is important, as it can dramatically reduce the entry barriers for the initial adoption of a solution. Solutions that can only be evaluated by multiple different user roles require a collective evaluation and adoption decision (CVI). **Trialability** was given for most solutions and is implemented in different ways. Some solutions provide free versions with usable but limited functionality, while more complex solutions have cloud-based trial versions with sample data included, and solutions with mobile interfaces provide the native apps for free (including demo mode) and only charge the backend system or service. Trials are mostly combined with video tutorials and documentation.

User-based pricing or subscription models are good solutions to provide a **starter package,** since the company can incrementally adopt an application without significant upfront investments. With regard to **minimal infrastructure**, cloud deployments are available even for the latest high-performance applications (e.g., S&OP). However, analytical big data applications, in particular, face an inherent issue with regard to infrastructure in that they mostly rely on data originating in other (often multiple different) systems and meaningful productive use is only possible if they are integrated (CVI). This issue was partly addressed by providing cloud-based integration services (S&OP) and standardized database drivers (SAP Lumira). Another option is to allow scenarios with a manual data uploads (e.g., via file upload,

SAP Lumira). The infrastructure and the effort for integration determine if an application can be **instantly used**. To reduce the implementation time, for instance, standardized service offerings are used (CVI). **Adoption by business users** is enabled if business users can evaluate usefulness and ease-of-use, and if they can decide on the adoption in a self-service mode using the online process. Notwithstanding the aforementioned measures, user interfaces that are oriented towards consumer-applications (e.g., Decisions) simplify the evaluation for users. The **e-commerce process** was rated as complete if sufficient information was accessible online to get informed and evaluate the solution (e.g., value descriptions, functional and technical documentation, video tutorials, pricing information), if trials could be accessed, and if the solution could ultimately be bought directly.

Limitations. Validity was taken into consideration during the design and execution of this research. The frameworks used to perform the case studies were developed using accepted theories and were tested with subject matter experts. Generalizability of case studies in general is limited [15], however, the measures identified may well be relevant for other enterprise applications. Reliability was addressed by describing the research process in as much detail as possible. Moreover, by applying the assessment model to multiple cases, theoretical replication was already inherent to the study. Lastly, criticisms maybe raised regarding the organizational context of this work. This was addressed by strictly adhering to the research process, involving subject matter experts for interpretative tasks, and presenting different points of view.

6 Conclusion

In this work, applications have been studied with regard to their readiness for the app store model. Multiple measures were identified as to how this readiness can be accomplished. Moreover, the study evaluated the applications based on the given e-commerce capabilities of today's app stores or online software sales channels. New e-commerce instruments may very well enable sales of applications, which, in today's environment are not well suited for an online channel (e.g., group evaluation). Furthermore, the online sales channel can be integrated closely with traditional personal, direct sales channels to "best cross-fertilize each other and exploit their respective strengths" [21]. Lastly, new innovations, such as the big data applications presented in this study, are relatively new to the market. It is to be expected that as they mature, the need for personal consultation will be reduced and standardization and app store readiness will increase.

References

1. Rapaport, L.: Apple Says Application Downloads Have Exceeded 40 Billion, http://www.bloomberg.com/news/2013-01-07/apple-tops-40-billion-app-downloads-with-half-in-2012.html
2. Weiß, F., Leimeister, J.M.: Consumerization - IT Innovations from the Consumer Market as a Challenge for Corporate IT. Bus. Inf. Syst. Eng. 4, 363–366 (2012)
3. Niehaves, B., Köffer, S., Ortbach, K.: The Effect of Private IT Use on Work Performance-Towards an IT Consumerization Theory. Wirtschaftinformatik Proc. 2013, 39–53 (2013)
4. Carr, N.: IT doesn't matter. Harv. Bus. Rev. (2003)

5. Vizard, M.: CIOs Struggle With Relevance of Role to Business, http://www.cioinsight.com/it-management/cios-struggle-with-relevance-of-role-to-business
6. Armbrust, M., Joseph, A.D., Katz, R.H., Patterson, D.A.: Above the Clouds: A Berkeley View of Cloud Computing. Science 53 (80-), 7–13 (2009)
7. Jennings, C.: The end of the IT department – is it in the cloud?, http://www.computerweekly.com/feature/The-end-of-the-IT-department-is-it-in-the-cloud
8. Verville, J., Halingten, A.: A six-stage model of the buying process for ERP software. Ind. Mark. Manag. 32, 585–594 (2003)
9. SAP: Simple UI for SAP Applications: SAP Fiori Improves User Experience, http://en.sap.info/sap-fiori-improves-user-experience/98278
10. Böckle, R.: B2B App Stores - Anbieter im Vergleich. Computerwoche 49, 14–21 (2013)
11. Novelli, F., Wenzel, S.: Adoption of an Online Sales Channel and "Appification" in the Enterprise Application Software Market. In: ECIS 2013 Proceedings (2013)
12. Pettey, C.: Gartner Identifies the Top 10 Strategic Technology Trends for 2013 (2013), http://www.gartner.com/newsroom/id/2209615
13. Pettey, C.: Gartner Identifies the Top 10 Strategic Technologies for 2012 (2012), http://www.gartner.com/newsroom/id/1826214
14. Luftman, J., Derksen, B.: Key issues for IT executives 2012: Doing More with Less. MIS Q. Exec. 11, 207–218 (2012)
15. Bryman, A., Bell, E.: Business Research Methods. Oxford University Press, USA (2011)
16. SAP: SAP Store, http://www.sapstore.com
17. SAP: SAP Big Data Business Applications, http://www.sapbigdata.com/applications
18. Harris, J., Ives, B., Junglas, I.: IT consumerization: when gadgets turn into enterprise IT tools. MIS Q. Exec. 11, 99–112 (2012)
19. Jones, D., Behrens, S., Jamieson, K., Tansley, E.: The Rise and Fall of a Shadow System: Lessons for Enterprise System Implementation. In: ACIS 2004 Proceedings (2004)
20. Beimborn, D., Palitza, M.: Enterprise App Stores for Mobile Applications - Development of a Benefits Framework. In: AMCIS 2013 Proceedings (2013)
21. Novelli, F., Wenzel, S.: Online and Offline Sales Channels for Enterprise Software: Cannibalization or Complementarity? In: ICIS 2013 Proceedings (2013)
22. Davis, F.D.: Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology. MIS Q. 13, 319–340 (1989)
23. Venkatesh, V.: Determinants of Perceived Ease of Use: Integrating Control, Intrinsic Motivation, and Emotion into the Technology Acceptance Model. Inf. Syst. Res. 11, 342–365 (2000)
24. Cyr, D., Head, M., Ivanov, A.: Design aesthetics leading to m-loyalty in mobile commerce. Inf. Manag. 43, 950–963 (2006)
25. Yin, R.K.: Case Study Research: Design and Methods (Applied Social Research Methods). SAGE Publications, Inc. (2013)
26. Stähler, P.: Geschäftsmodelle in der digitalen Ökonomie. Merkmale Strategien und Auswirkungen. Josef Eul Verlag (2002)
27. SAP: SAP HANA Marketplace, http://marketplace.saphana.com
28. SAP: SAP Application Development Partner Center, http://www.sapadpc.com
29. SAP: SAP Homepage, http://www.sap.com
30. SAP: SAP HANA Portal, http://www.saphana.com
31. SAP: SAP Community Network, http://scn.sap.com
32. SAP: SAP Help Portal, http://help.sap.com
33. Liquid-Analytics: Liquid Analytics Homepage, http://www.liquidanalytics.com

Temporal Reconfiguration-Based Orchestration Engine in the Cloud Computing

Zaki Brahmi[1,3] and Chaima Gharbi[2,3]

[1] Higher Institute of Computer Science and Communication Technique of Hammam Sousse, Sousse University, Tunisia
zakibrahmi@yahoo.fr
[2] Higher School of Business, Manouba University, Tunisia
gharbi_shaima@live.fr
[3] RIADI-GDL Laboratory, Manouba University

Abstract. In our days, the cloud computing wins a great importance. So it becomes the refuge of many companies especially Small and Medium sized enterprises (SMEs), since it provides computer services witch fits with demand and charged according to their use. Now the evolution towards the cloud is promoting that orchestration business process to be run as a service (Orchestration as a Service (OaaS)). OaaS represents a solution especially for (SMEs) which needs IT Systems intergration, but cannot install and use such integration platforms because of their maintenance costs and operation. OaaS is a specialization of paradigm Platform as a Service (PaaS). It reduces integration costs by outsourcing the operation and maintenance of an orchestration engine to an OaaS provider. The orchestration engine must be able to maintain its functionalities and performances in case of high demand. It has to be faster and the users have to pay less to run their orchestration processes. In this article, we propose an orchestration engine as a service based on the temporal reconfiguration approach. The proposed approach is based on two main ideas : i) Partition the amount resources of cloud server proportionally between BPEL processes. ii) Applying a temporal partitioning algorithm on a set of BPEL process. Our approach can be executed in a dynamic environment and is scaled with the number of BPEL processes.

Keywords: Cloud Computing, Orchestration, Temporel partitioning, BPEL, Reconfigurable computing system.

1 Introduction

The Cloud Computing has recently emerged as a compelling paradigm for managing and delivering services over the Internet. The rise of Cloud Computing is rapidly changing the landscape of information technology, and ultimately turning the long-held promise of utility computing into a reality [1]. The Cloud Computing offers three basic services: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). All these can be deployed through private, public or hybrid cloud deployment modules. According to the

W. Abramowicz and A. Kokkinaki (Eds.): BIS 2014, LNBIP 176, pp. 73–85, 2014.

National Institute of Standard and Technologies, the Cloud is revolutionizing the way companies run their services ; the pay-per use paradigm has proven an effective way to reduce Information technology (IT) costs without sacrificing quality, which is attracting an increasing number of companies [2].

The integration represents a challenge for many enterprises, in order to allow transparent interaction (heterogeneous) systems and computer services. Unfortunately, the applications are not generally easy to integrate for many reasons (the technology on which they are based are different, their API [1] are not compatible, etc.) [3]. Modern integration solutions adopted the service oriented architecture (SOA) design paradigm. In order to execute integration platforms applicable to SOA integration, costs such as licenses, hardware, and system administrators have to be taken into consideration [4]. A lot of companies like small and medium enterprises (SMEs) are not able to exploit such integration platforms although they certainly have intergration needs. The Cloud Computing provides a solution that satisfies this needs ; Orchestration as a Service (OaaS).

Orchestration as a Service (OaaS) is a new concept that appears in Cloud Computing, which denotes that the orchestration engine is hosted in a cloud environment, directly to be maintained by the OaaS provider. It is a specialization of paradigm Platform as a Service (PaaS). OaaS represents a solution for many companies, especially Small and Medium Enterprises (SMEs) which have IT systems integration needs, but cannot install and use such integration platforms because of their maintenance costs and operations. It reduces integration costs by outsourcing the operation and maintenance of an orchestration engine to an OaaS provider [4] [5]. OaaS requires very efficient scalable orchestration engines ; the less customers must pay, the more they can be served. An orchestration process can be described using a variety of languages, including WS-BPEL. The Business Process Execution Language (BPEL) [6] is a standard executable language that can be used to specify composite web services.

Few works have been proposed in the literature to resolve the problem of devising efficient orchestration engines: The Bis-Grid Engine [4] and The Guaran RT [5]. However these engines suffers from scalability and dynamicity.

In this paper we propose an orchestration engine based on the temporal reconfiguration approach. This approach is based on two main steps: i) Partitioning the amount resource of cloud server proportionally between BPEL processes. ii) Applying a temporal partitioning algorithm on a set of BPEL process. The temporal partitioning algorithm used in this work is based on the reconfigurable architecture that has gained much popularity among researchers and scientists because of their flexibility and high performance. This algorithm is based on the eigenvectors of the graph to find the best schedule of nodes that minimizes the communication between design partitions of the graph.The temporal partitioning algorithm can be very useful in the Cloud field, since it can be adaptable to the applications changes due to the reconfigurable architecture.

The remainder of this paper is organized as follow: in section 2, we will make a study of the related work. We will focus on Section 3 on the basic used concepts

[1] API (Application Programming Interface).

and the temporal reconfiguration formulation problem. We will describe our proposed orchestration engine in section 4. Section 5 will shows the evaluation of our approach. Finally, we will conclude our work and we refer to our prospects.

2 Related Work

There are only few numbers of recent articles in which the authors have paid attention to the problem of designing efficient orchestration engines. In the following section, we will present some of these projects, but also regard related work concerning the partitioning of BPEL program.

2.1 Orchestration Engines

In [4], the authors present the Bis-Grid project. Its objective is to combine the standard WS-BPEL for orchestration services and grid technologies with comprehensive security mechanism to provide secure integration platforms as Orchestration as a service. The BIS-Grid Engine is based on the UNICORE 6 which is a Grid middleware that provides comprehensive security mechanisms. The key concept of BIS-Grid Engine is to use an arbitrary WS-BPEL engine for workflow execution encapsulated by service extensions to UNICORE 6. However, this engine suffers from scalability which is a very important feature in the field of Cloud Computing. In fact, for each service, the Bis-Grid engine creates an instance in the ActiveBPEL engine which affects in turns a thread for each instance. If the number of services increases, then the number of threads increases, therefore the engine becomes very cumbersome and the response time increases. So, the Bis-Grid engine can not be efficient for a large number of services.

In [5], the authors propose an orchestration engine called Guaran RT. This engine relies on effective implementation that uses a configurable thread pool that works at the granularity of tasks; Instead of threads allocating to process instances, it allocates threads to individual tasks inside processes. The only missing thing in Guaran RT is how it can work in an elastic environment (i.e., how it can be flexible enough, to handle the changes that can affect certain services in the Cloud).

In [7], the authors annonce an orchestration engine which focuses on services orchestration that are distributed across wide-area networks. In [8], the authors focus on the increasing of productivity in cloud computing environments by minimise the scheduling of tasks. In [9], the authors specifie the technique of the selection of workflow tasks optimally in cloud environments.

In Order to solve these problems, we choose to use the partitioning technique of BPEL process in the orchestration engine. Thus, we present in the next section some work related to the partitioning of BPEL program.

2.2 BPEL Partitioning Approaches

In [10], the authors propose a partitioning approach based on genetic algorithm. To apply the genetic algorithm the tasks of BPEL program are represented by a

genome. However, the genetic algorithm relative to other meta-heuristics requires extensive calculations. Besides, genetic algorithm in this article uses two types of tasks; fixed tasks and portable tasks. Knowing that the fixed tasks are executed at providers and portable tasks can be executed at any server. If we have an issue with a provider or at a portable or fixed task, then will have to repeat the entire algorithm and this will increase the response time.

In [11], the authors propose a new partitioning algorithm for BPEL processes. The aim of this algorithm is to determine the best site where each portable task has to be performed in order to optimize the overall performance of the orchestration. The authors are based on the fusion along def-use chains which is very interesting because it thins the space of all possible merging. This approach cannot support a large number of web services since it is used in a LAN (Local Area Network) environment.

In [12], the authors have proposed a new temporal partitioning algorithm for an embedded system. This algorithm is divided on two main steps. The first step aims to find an initial partitioning of the graph; in order to minimize the transfer of data required between design partitions. The second step aims to find the final partition of the graph while satisfying the constraint area. Then, the temporal partitions are configured one after another on the reconfigurable device (FPGA [2]). Due to the FPGA, the temporal partitioning algorithm can be adapted to necessary changes; this means that we can incorporate the changes without losing any time to resume the entire algorithm. Indeed, in an FPGA each processing task runs in full autonomy without depending on other tasks.

In [13], the authors have proposed a framework for parallelizing service composition algorithms investigating how to partition the composition problem into multiple parallel threads.This framework is based on the technique "Binary_Tree", which is a representative model promising reasonable efficiency for composition algorithms. Since, the straightforward parallelization techniques do not lead to superior performance, the authors in [13] propose two new approaches to evenly distribute the workload in a sophisticated fashion. Theoretically, the parallelization service composition algorithm is very complicated which makes the operation hardly feasible.

3 Problem Statement

The temporal reconfiguration problem of BPEL process *(TRPP)* is to run a set of BPEL process P on a cloud server S while respecting its capacity C. The problem *(TRPP)* can be presented by a couple $TRPP = (C, P)$ where C is the resource capacity of the server S, and $P = \{p_1, p_2 \dots p_n\}$ represents the set of BPEL processes to be run in the server S. Each BPEL process P_i admits a cost $cost\,(P_i)$, which represents the amount of resource requested by the process P_i .

Given a set of BPEL processes running on a server S, the problem arises when there is a small insufficient ressource for the execution of a new process.

[2] FPGA is a semi-conductor device which is based on a matrix of logic blocks recongurables connected through programmable interconnections.

So the temporal reconfiguration problem of BPEL process is to partition the new process intol sub-partitions, in order to run each time a sub-partition in this small insufficient ressource.

Definition 1 (BPEL process) : is a quadratic $BPEL = (PL, V, O, cost)$.Variables V are used to define data relating to the internal state of the process, to store, format and transform messages with web services invoked. Orchestration O gives the definition of the process in terms of the different types of activities $A = \{Receive, Reply, Invoke, Send. . . . Assign\}$ that offer BPEL. The cost represents represents the amount of resource requested by a process.

We propose the following formula to calculate the cost of a BPEL process P_i:

$$cost(P_i) = \sum_{a \in A} Num_a \times cost_a \tag{1}$$

Where Num_a represents the number of activity a in the process P_i and $cost\,(a)$ denotes the cost of each activity $a \in A$

Definition 2 (Temporal Partitioning): A temporal partitioning of $G = (E, V)$, is its division into k disjoint partitions $P = \{P_1, P_2, \ldots, P_k\}$. A temporal partitioning is feasible in accordance to a reconfigurable device H with area A(H) and pins T(H) (number of programmable input/outputs (I/Os) per device); if the two conditions are verified [12]:

$$C1 : \forall P_i \in P; A(P_i) \leq A(H) \tag{2}$$

$$C2 : TCCost = \sum_{i=1}^{k} CCost\,(P_m) = \sum_{m=1}^{k} \sum_{T_i \in P_m; T_j \in \bar{P_m}} \alpha_{i,j} \leq T\,(H) \tag{3}$$

Where $TCCost$ denotes total communication cost, $CCost(P_m)$ denotes communication cost of the partition P_m and the weight $\alpha_{i,j}$ of an edge $e_{i,j}$ defines the amount of data transferred from T_i to T_j .

Definition 3 (Reconfigurable computing): Reconfigurable computing as a concept existe since quite some time. Reconfigurable computing systems represent a new type of computer architecture that combines a flexible part of software with high performance of hardware. It is a new paradigm to meet the simultaneous demand for efficient and flexible application. Reconfigurable computer systems involve the use of reconfigurable devices such as FPGAs [14] .

4 Orchestration Engine as a Service Based-Temporal Reconfiguration Approach

In this section, we will present our orchestration engine based on the temporal reconfiguration approach. Our orchestration engine has to:

- *Exploit all available resources to execute a maximum number of BPEL processes*: This criterion allows exploiting small ressources to run the new process.

- *React in a dynamic environment*: In fact, the dynamic aspect is an important criterion in the field of web services.
- *Reply to a maximum number of users' queries*: It has to be able to handle multiple user requests simultaneously.
- *Reduce the communication cost*: this criterion reduces the communication cost between the different sub-processes.
- *Minimize the response time of the user's requests*: Based on the technique of temporal partitioning of BPEL processes into sub-processes, our orchestration engine can reduce the response time of the user's requests. Indeed, our engine exploits small amounts of resources to execute the sub-processes in order to eliminate the waiting time to find a sufficient amount of resources to perform the new processes.
- *Scalable* : The orchestration engine has to be able to maintain its functionality and performance in case of high demand.

In the next section, we will describe our approach at first. Next, we will describe the application of the proposed approach on a case study.

4.1 The Characteristics of the Proposed Approach

Our approach is based on two ideas :

- Partitioning the amount of resource proportionally between BPEL processes ; this step can exploit all the resources offered by the server in order to run a maximum number of BPEL processes.
- Applying a temporal partitioning on a set of BPEL process ; this step is used to divide a BPEL process into sub-processes, in order to quickly perform all customer requests.

Partitioning the Amount of Resource between BPEL Processes: To run different BPEL processes on a server with capacity C, we need to partition proportionally this capacity between the different processes. To achieve this part, we propose the above formula :

$$CP_i = \frac{\frac{C}{n} \times cost\,(P_i)}{\sum_{i=1}^{n} cost\,(P_i)} \tag{4}$$

Where :

- CP_i denotes the amount of resource proposed by the server S for each process P_i
- C represents the capacity of the server S
- n : represents the number of BPEL process to be executed in the server S
- $cost(P_i)$: represents the cost of each process

In order to calculate the amount of resource assigned to each BPEL process and efficiently exploit the amount resource of cloud server S, we propose the algorithm (Calculate_QRessource_PBPEL) which allows to partition proportionally this amount of resource among a set of BPEL processes.

Pseudo-code : Calculate_QRessource_BPEL

Input: $P = \{P_1, P_2, \ldots, P_n\}$, $C//$ the amount ressource of server
Output : $T\,[i]$
$Sum\_cost = 0$
Begin
for each process $P_i \in P$ **do**
$Tab\,[P_i] \leftarrow calculate\_cost\,(P_i)$
$Sum\_cost \leftarrow Sum\_cost + Tab\,[P_i]$
end for
for each process $P_i \in P$ **do**

$$CP_i \leftarrow \frac{\frac{C}{n} \times Tab\,[P_i]}{Sum\_cost}$$

$T\,[i] \leftarrow CP_i$
end for
End

The function $calculate\_cost\,(P_i)$ calculates the cost of each process, Sum_cost represents the sum of the costs of n processes, and $Tab\,[P_i]$stores the cost of each process. $T\,[i]$ stores the different values of CP_i

Applying a Temporal Partitioning on a Set of BPEL Process: To partition a BPEL process, we have based on the work presented in [12], since this algorithm reduces the communication cost, ie reduces the use of memory as compared to other algorithms. This approach can be useful, because the embedded system and the cloud computing have a unique feature that they work in real time. In addition, this algorithm is based on reconfigurable architecture which became more popular among researchers and scientists due to their flexibility and high performance. The temporal partitioning algorithm is used to find a way of partitioning the graph with an optimal number of partitions where the communication cost has the lowest value with respecting all constraints. To apply the temporal partitioning algorithm for each BPEL process, we have to go through two steps. The first step is to find an initial partition P_{in} of the graph. This step gives an optimal solution in terms of communication cost. Then, if the resource constraint is satisfied after the first stage, we adapt the initial partitioning, else we move to the second stage. The purpose of the second step is to find the final partition P of the graph while satisfying the resource constraint. If the second stage cannot find a feasible schedule, then we reduce the number of stages by one and the algorithm goes to the first step, and then we restart to find a feasible solution with the new number of stages.

a) The intermediate representation

In [12], the authors use as input a Data Flow Graph. However, the BPEL process is a specific programming language, so we have to use a BPEL Flow Graph in order to treat effectively the characteristics of BPEL. BPEL Flow Graph (BFG) is an extension of Control Flow Graph [16] which is used to represent a BPEL program in a graphical mode. BFG does not only contain information structure, which specifies all the information about control flow of BPEL program and data flow information, but also semantic information such as dead paths. The structural definition of the BFG is as follows:

$$BFG =< N, E, s, F >$$

where :

- N is a set of nodes
- E is a set of edges
- s is the start node
- F is a set of endpoints

b) The communication cost

The primary goal of temporal partitioning algorithm is to find a way to partition a graph where the communication cost has the lowest value. So to build optimized partitions we have to estimate the communication cost between pairs of activities of a BPEL process. The authors in [15] assume that the communication between activities can be measured by bytes. Also, they consider that a control message has a size of one byte, and the average size of a data message type can be known, and is denoted *taille (d).* So to determine the number of bytes that would be exchanged between the services assigned to the a_1 and the service assigned to a_2, we need to calculate : 1) How many times a given activity will be executed. We write $numExec(a)$ to denote this amount. 2) Given two consecutive activities a_1 and a_2, what is the probability that one execution of activity a_1 is immediately followed by an execution of activity a_2. We write $probExec(a_1, a_2)$ to denote this probability.

The authors in [15] define the cost of communication $co\,(a_1, a_2)$, between two activities a_1 and a2 :

$$co\,(a,b) = cons\,(a,b) \times nbExec\,(a) \times probExec\,(a,b) + \tag{5}$$

$$+ \sum_{(a,b,d)\epsilon\ Data} nbExec\,(a) \times Taille\,(d)$$

Where $cons\,(a_1, a_2)$ is a function equal to one if a_1 and a_2 are consecutive activities, and zero otherwise. The first term in this formula corresponds to the communication overhead induced by control-flow notifications, while the second term corresponds to the communication overhead induced by data-flows.

The algorithm (Temporel_Partitioning_BPEL) presents the temporal partitioning for a set of BPEL processes.

Pseudo-code : Temporel_Partitioning_BPEL

Input : $P \leftarrow \{P_1, P_2, \ldots, P_n\}$
Output : $PG \leftarrow \emptyset$
Begin
for each process $P_i \in P$ **do**
  $G_i \leftarrow transform\_BFG\,(P_i)$
  for each node $a \in G_i$ **do**
    for each node $b \neq a$ **do**
      if there is a path between a and b **then** **then**

$$co\,(a,b) \leftarrow cons\,(a,b) \times nbExec\,(a) \times probExec\,(a,b) + \sum_{(a,b,d) \in Data} nbExec\,(a) \times Taille\,(d)$$

      end if
    end for
  end for
  $PG \leftarrow PG \cup partition\_temp\,(G_i)$
end for
End

The fonction *transform_BFG(P_i)* transforms each process P_i into a BPEL Flow Graph *Gi*. The fonction *partition_temp*(G_i) partitions each process G_i into temporal partitions using a temporal partitioning algorithm. *PG* used to store all the graphs that have been partitioned.

4.2 Example

To understand better the functioning of our approach, we represent this example. Suppose that the server S admits a resource capacity C = 20. At the instant t_0, the server runs 11 BPEL process and a new BPEL process comes to be executed in the server S (Fig. 1)

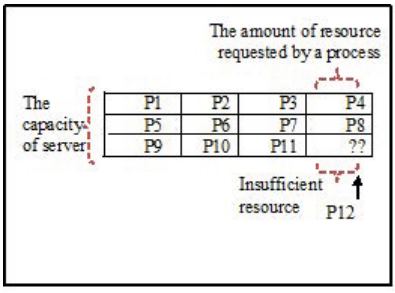


Fig. 1. The Server S

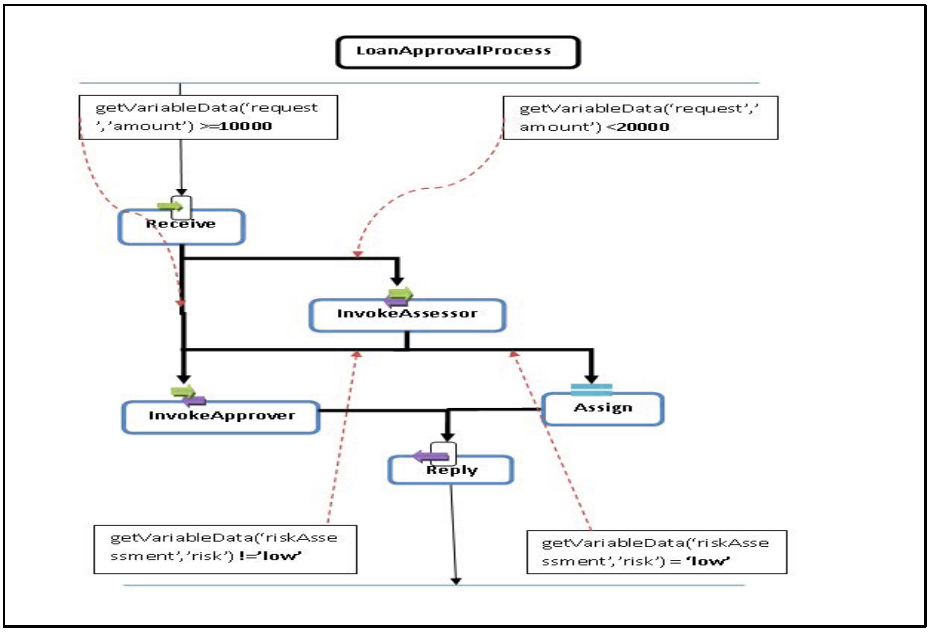
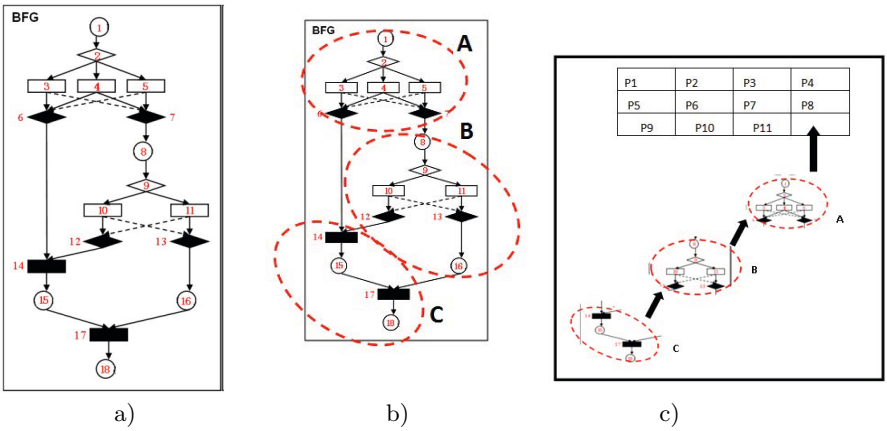


Fig. 2. Example of BPEL program of Loan approval process[16]

This new process is presented in(Fig. 2), which is an example of a BPEL program describing a loan approval process.

This process begins by receiving a loan request. For low amounts (less than $20,000) and low-risk individuals, approval is automatic. For high amounts or highrisk individuals, each credit request needs to be studied in more details. The use of risk assessment and loan approval services are represented by invoke elements, which are contained within a flow, and their (potentially concurrent) behavior is staged according to the dependencies expressed by corresponding link elements. Note that the transition conditions attached to the links determine which links get activated, all the join conditions use default setting. Finally the process responds with either a "loan approved" message or a "loan rejected"message[16]. However, at this moment the server admits only a capacity that is equal to 5. This capacity is insufficient to perform this new process which admits a cost equal to 7. So we try to partition the process by applying the temporal partitioning algorithm. To apply it, the new process has to be converted into a graph. (Fig. 3(a)) shows the BPEL Flow Graph for the new process.

The figure (Fig. 3(b)) represents a partition of the BFG. The Partitioning obtained in (Fig. 3(b)) may undergo changes during the time, for example we can find a new partitioning that minimizes the communication cost more than the previous partitioning or the process may undergo changes during the time.



a) b) c)

Fig. 3. a) BPEL Flow Graph (BFG) for Loan approval process b) The partitioning of the BFG c) Execution of the BPEL process in the server S

So the temporal partitioning algorithm used has to be adaptable to modifications. Knowing that each partition is implemented as a separate circuit module which can be changed as necessary at run-time. Thus if there is a change in one of these partitions, we can modify it without changing the preceding partitions.

Now the server starts the execution of these sub-partitions one after another in order to run at the end the entire process, since the cost of each partition A, B and C has to be less or equal to 5 (Fig. 3(c)).

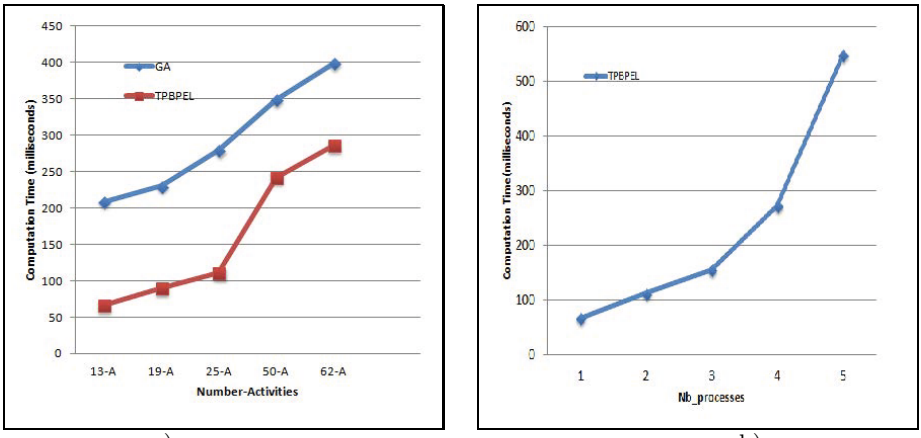
5 Implementation and Evaluation

The experimentations are performed on an Intel Core 2 Duo CPU T6600 380@2.53GHz with 2 GB of RAM. To implement our approach we use Netbeans 7.0 .

Firstly, we compare the performance of our temporal partitioning algorithm (TPBPEL) with other partitioning approaches based on genetic algorithm [10] and we record the computation time of each algorithm depending on the size of the process. The size of the problem depends on the number of activities in the BPEL process. The results of these experiments are presented in (Fig. 4(a))

In these experiments, the processes used have been named according to the number of activities in the workflow of BPEL program. For example, the process 13-A means that there are 13 activities in the workflow of its corresponding BPEL program. The (Fig. 4(a)), show two curves representing the evolution of the execution time depending on the sizes of process. The first curve shows the evolution of the execution time of our approach and the second represents the evolution of the execution time of the approach using genetic algorithm [10].

We can notice that the computation time of our approach increases slowly with the problem size. In contrast, the computation time of the genetic algorithm increases considerably with the the problem size.



a) b)

Fig. 4. a) Comparisons of the computation time of our algorithm (TPBPEL) versus Genetic Algorithm (GA) for resultant partitioning of different BPEL program sizes. b) Execution time of our approach with different process number.

For the process 62-A having a large size, our approach has spent 286 (ms) to find a solution while the genetic algorithm (GA) requires 399 (ms) to solve the problem.

The (Fig. 4(b)) show the evolution of the execution time of our approach based on the number of BPEL processes. For these experiments, we vary the number of processes between 1 and 5 processes.

6 Conclusions and Futre Work

In our present work, we propose a new orchestration engine based on the temporal reconfiguration approach. Our solution is divided on two steps: i) Partitioning the amount of resource proportionally between BPEL processes. ii) Applying a temporal partitioning algorithm on a set of BPEL process. This approach is based on a reconfigurable architecture which provides to our engine a dynamic aspect. In our future works,we plan to add a security mechanism and to use the multiCloud instead of working with a single cloud in order to improve our approach.

References

1. Qi, Z., Lu, C., Raouf, B.: Cloud computing: state-of-the-art and research challenges. Journal of Internet Services and Applications, 7-18 (2010)
2. Lee, B., Tim, G., Robert, C.P., Voas, J.: DRAFT Cloud Computing Synopsis and Recommendations. Recommendations of the National Institute of Standards and Technology (2011)

3. Rafael, F., Reina, Q.: A Domain-Specific Language to Design Enterprise Application Integration Solutions. International Journal of Cooperative Information Systems (IJCIS),143–176 (2011)
4. Höing, A., Scherp, G., Stefan, G.: The BIS-GRID Engine: An Orchestration as a Sservice Infrastructure. International Journal of Computing, 96–104 (2009)
5. Frantz, Z., Rafael, C., Arjona, L.: An Efficient Orchestration Engine for the Cloud. In: 4th IEEE International Conference on Cloud Computing Technology and Science Proceedings, pp. 711–716 (2011)
6. Juric, M.B.: Business Process Execution Language for Web Services BPEL and BPEL4WS. Packt Publishing (2006)
7. Yang, H., Kim, M., Karenos, K., Ye, F., Lei, H.: Message oriented middleware with QoS awareness. In: ICSOC (2009)
8. Grounds, N.G., Antonio, J.K., Muehring, J.: Cost-minimizing scheduling of workflows on a cloud of memory managed multicore machines. In: Jaatun, M.G., Zhao, G., Rong, C. (eds.) Cloud Computing. LNCS, vol. 5931, pp. 435–450. Springer, Heidelberg (2009)
9. Song, B., Hassan, M.M., Nam Huh, E.: A novel heuristicbased task selection and allocation framework in dynamic collaborative cloud service platform. In: CloudCom, pp. 360–367 (2010)
10. Ai, L., Tang, M., Fidge, C.: Partitioning composite web services for decentralized execution using a genetic algorithm. Future Generation Computer Systems, 157–172 (2011)
11. Nanda, M.G., Satish, C., Sarkar, V.: Decentralizing Execution of Composite Web Services. ACM, 170–187 (2004)
12. Ramzi, A., Bouraoui, O., Abdellatif, M.: A partitioning methodology that optimizes the communication cost for reconfigurable computing systems. International Journal of Automation and Computing, 280–287 (2012)
13. Hennig, P., Balke, W.: Highly Scalable Web Service Composition Using Binary Tree-Based Parallelization. In: IEEE International Conference on Web Services, ICWS 2010, Miami, Florida, USA, pp. 123–130 (2010)
14. Compton, K., Hauck, S.: Reconfigurable Computing: A Survey of Systems and Software. ACM Computing Surveys, 171–210 (June 2002)
15. Walid, F., Marlon, D., Godart: Heuristics for composite Web service decentralization. Software and Systems Modeling, 1–21 (2012)
16. Yuan, Y., Li, Z., Sun, W.: A Graph-Search Based Approach to BPEL4WS Test Generation. In: International Conference on Software Engineering Advances, p. 14 (2006)

Data State Description for the Migration to Activity-Centric Business Process Model Maintaining Legacy Databases

María Teresa Gómez-López, Diana Borrego, and Rafael M. Gasca

Departamento de Lenguajes y Sistemas Informáticos,
Universidad de Sevilla, Spain
{maytegomez,dianabn,gasca}@us.es
http://www.lsi.us.es/~quivir

Abstract. One of the reasons why the companies keep out the business process adaptation, is focused on the complexity to adequate their databases to a Business Process Management. It implies to determine the relation between the activities of the process, and the data objects stored in the database. Our proposal allows the business expert to know the state of the data objects according to the business process, facilitating the migration.

In this paper, we propose a methodology and a set of mechanisms to support the data adaption, from a legacy database to an activity-centric business process. This methodology lets the description of the data object states, and their equivalences with the relational database by means of an Object-Relational Mapping and a Domain Specific Language.

Keywords: Business Process, Integration of data and processes, Data object State, BP Migration, Object-Relational Mapping.

1 Introduction

The benefits of employing an Activity-Centric Business Process Management are numerous [1]. Among the benefits are: (i) it is possible to choreograph in a single process tasks implemented using different technologies; (ii) the business expert can help in the process specification, since graphical languages have been defined to be understandable; (iii) a business model can help to reduce overheads, as a result of less time or resources being used for an end result; (iv) it is easier to detect process inconsistency and optimise it; and (v) there are several tools and technologies that let to improve the business intelligence. The problem is that organizations currently need to manage a great deal of data. When a company wants to incorporate a Business Process Management System (BPMS) to improve their functionality, all these data have to be taken into account.

Some of the stored data represent the values involved in the business process execution, although they had not been included in the database by means of a BPMS. The execution of an activity of a business process, can imply the modification of the state of a data object, understanding the database as a

W. Abramowicz and A. Kokkinaki (Eds.): BIS 2014, LNBIP 176, pp. 86–97, 2014.

repository of business data objects. The inconvenient is the existing gap between business data objects and relational data. The values of these stored data, also represent the state of each instance that is being executed in the business process of a company. For example, if an attribute of a project to represent the final date is null, it means that the project has not finished yet. The modification of the data (and the state) can be performed by means of a BPMS, a set of unconnected applications or human tasks. For this reason, our proposal provides a methodology and the necessary mechanisms to model and know the state of the data objects stored in a database to facilitate the migration to a business process. It also implies to know the last activity that was executed for each data object, and the state of the object. We propose an automatic executable migration process, based on the use of mature software technologies.

When a company has all their data in a relational database, and it wants to manage a part of these data by means of a BPMS, the next steps of the methodology are necessary. Initially, we count on an activity-centric business process model aligned with the relational database object of the migration.

1. **To include the data objects description into the business process model.** BPMN 2.0 [2] has incorporated the data description necessity, but some capacities are still missed, such as, the description of the data relation with the persistent information, or the state data description in accordance with its value.
2. **To describe the conceptual model in accordance with the relational model of the legacy database.** Relational model is very detailled, being difficult to understand and query by non-expert users. We propose the use of a Conceptual Model managed by means of an Object-Relational Mapping (ORM).
3. **To describe the state of each data object in function of the value of its attributes.** Currently, the state of the object can be described by means of string labels. Unfortunately, it is possible to find a legacy database that do not represent the state of the objects in this way, since the concept of state is derived from the business data object description. In this paper, we suppose that every stored objects are in a defined state, and in only one, planning for future work the verification of this supposition.

The next question is who performs each step. The first and third steps must be performed by the business expert, since (s)he knows how the business works. For this reason, we propose the definition of a high level language to describe the state of the business data objects, and how they are transformed automatically into an executable code. The second step implies technical knowledge about relational and conceptual models, together with Java implementation capacity, being necessary a programmer to perform it.

This paper is organized as follows. Section 2 introduces the existing work about data object modeling in activity-centric business processes. Section 3 presents a motivating example to illustrate the concepts. Section 4 explains how the use of ORM can help in the business data object manipulation. Section 5

presents our proposal about how to adapt ORM and the annotations over the attributes of the business data objects, to reflect and implement the state of the object. Section 6 analyses an overview of related work found in the literature. And finally, conclusions are drawn and future work is proposed in Section 7.

2 Data Object Modeling in BPMN

In order to support the first step of the methodology, it is necessary to analyse the existing proposals about data object description in business processes. Although BPMN [2] is not primarily designed for data modeling, there is still a set of notations that lets you model the data involved in a business process. The primary construct for modeling data within the process flow is the `DataObject` element. A `DataObject` has a well-defined lifecycle, with resulting access constraints. Data Object References are a way to reuse Data Objects in the same diagram. They can specify different states of the same Data Object at different points in a process. Data Object Reference cannot specify item definitions, and Data Objects cannot specify states. The names of Data Object References are derived by concatenating the name of the referenced Data Object and the state of the Data Object Reference in square brackets as follows:

<Data Object Name> [<Data Object Reference State>]

The idea is that a process activity reading a data object, may only get enabled if that data object is in a particular state; when the activity is executed, the object may transit to a new state.

Figure 1 represents an example of accesses to the data objects X and W from an activity A. A data flow edge from a data object to an activity describes a read access to an instance of the data object, which has to be presented in order to execute the activity. Likewise, a data flow edge from an activity to a data object describes a write access, which creates a data object instance, if it did not exist (labelled with *[new]* [3]), or updates the instance, if it existed before. Also, a data flow edge connecting a data object with a flow in the business process model indicates the data object which is flowing through that connector, and the state of the flowing data object. Data objects can be modeled as a single instance or as a multi instance (indicated by three parallel bars) that comprises a set of instances of one data object. In accordance with this idea of representation, Figure 1 shows the read access to a multiple instance of the data object X, which should be in the state Y, and the write access to the data object W, which transition to the state Z after the execution of A. Likewise, after the execution of activity A, an XOR is executed, being the data object V in state U the one that flows through the upper branch.

We are inspired by the contribution in [3] that proposes an extension of BPMN, and the possibility to include stored data. Analysing this proposal, some considerations are necessary:

- They consider the attributes to distinguish data objects (i.e. primary keys) as a part of the data object representation in the business process model. Likewise, they also include the foreign keys in the representation, in orderto

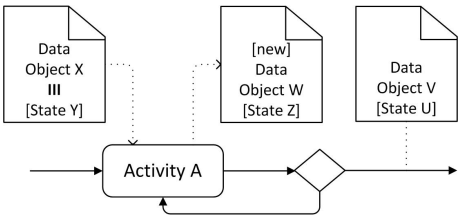


Fig. 1. Example of access to a data object

express the attributes to refer to the identifier of another object. We consider the necessity of a higher level of abstraction to represent data objects, to avoid that the business modeler needs to know about these database modeling details.

– In [3], the state of each data object is considered as a string stored within a column of the corresponding table in the database. This is very restrictive, since it reduces the scenarios to those where this column exists in the database, which is not very common. Instead, it would be desirable to infer the state of the data objects from various attributes of the database content.

3 Motivating Example

In this paper we have used an example of the paperwork of the presentation of a thesis. The example shown in Figure 2 is a simplification of the real process described by the University of Seville.

The business process model presented includes the creation and modification of states of the data objects which are handled during the process. The process begins with the application of a thesis project by the Student, creating the object *ThesisProject*. After that, the project documents are sent to the Department, which studies the documentation (the object *ThesisProject* changes its state to *evaluated*), and informs about this evaluation to the Student. If the thesis project is denied, the Student should repeat the previous steps. In case the *ThesisProject* is *approved*, the Student should finish the document of the thesis (creating the object *Thesis*) and send the documents to the Department for their deposit. After the time of deposit has finished and the committee have been evaluated and approved, a department meeting is celebrated to evaluate all theses which have been prepared in the department, since the last department meeting. Then, the Student is informed about this evaluation and can present the thesis in case the evaluation is positive.

We suppose that the relational model of Figure 3 supports the data objects of the thesis business process. Since relational model has several details, we propose the use of the conceptual model (Figure 4). The use of the conceptual model simplifies, for example, the use of auxiliary tables, such as *Supervisors*, *EvaluationCommittee* and *SubstitutionCommittee*, in order to model the n:m relationships.

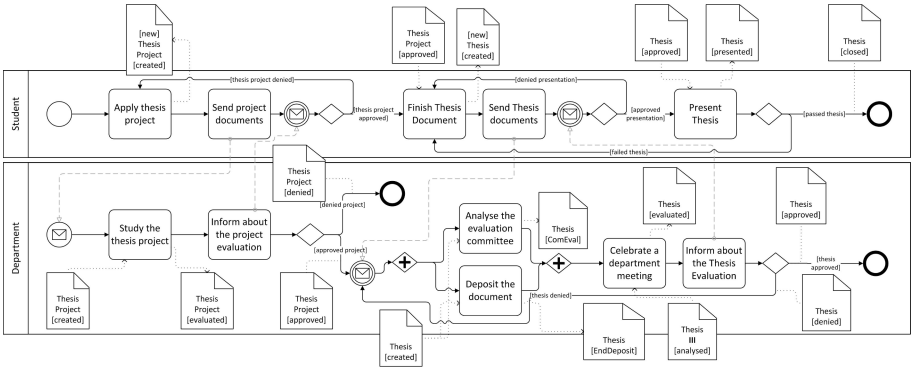


Fig. 2. Paperwork Thesis Process Example

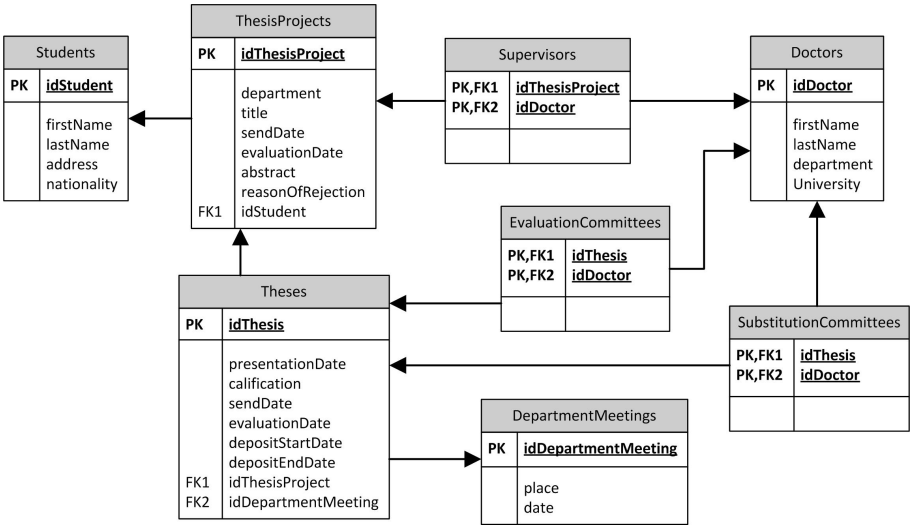


Fig. 3. Relational model of the example

4 Object-Relational Mapping to Represent Business Data Objects

In order to support the second step of the methodology to describe the conceptual model in accordance with the relational model, we need to determine the mapping between both models. One of the problems to represent data persisted in a relational database as objects is the complexity of the object relations, and how they are loaded and stored. Currently, in the development of software, the description of the data layer is not oriented to the relational model, but

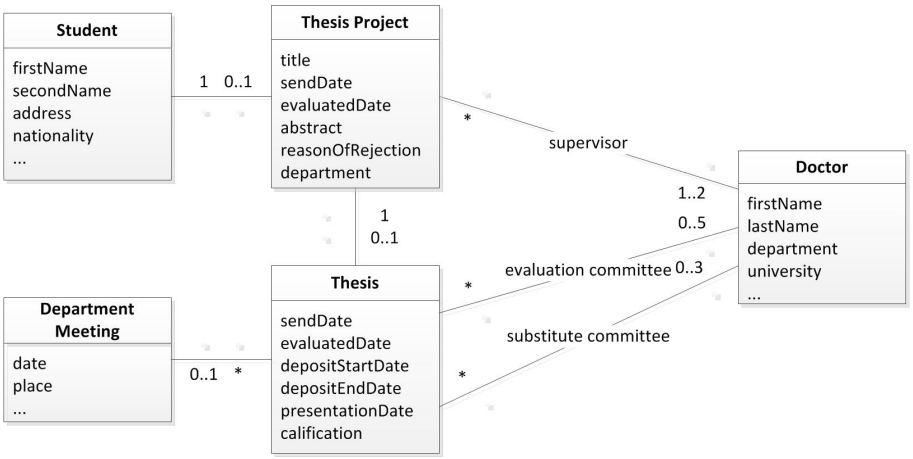


Fig. 4. Conceptual model of the example

the conceptual model is used instead. Common Data Access Object (DAO) implementations are provided by Object-Relational Mapping (ORM). ORM [4] is a programming technique for converting data between incompatible type systems in object-oriented programming languages. The use of ORM brings great benefits, as database independence, low coupling between business and persistence, and fast software development. This creates, in effect, a "virtual object database" that can be used from within the programming language. We think that this paradigm fits to the data objects in business processes, since it allows the designer to describe the data evolution in a more abstract level. One of the most used frameworks in ORM is Hibernate [5] whose annotation permits to define the relations between the tables and the data objects. For example, the annotation `@Id` in an attribute of a class describes that this represents the primary key of the table, or by means of `@Column` that the attribute represents a column. Describing the mapping between the tables and the classes, it is possible to save an object in a table of the database (persist method), remove instances from the database table (removed method), and get existing objects from the database (find method). These methods, and the ones defined for each class, isolate the business modeler to know the relational structure of the data stored and managed in the business process.

In this paper, only the necessary annotations and structures for the object data representations in Hibernate are introduced. In order to better understand the use of ORM, and how it solves all the types of relationships between objects (1:1-@OneToOne, 1:n-@OneToMany, n:1–@ManyToOne, n:m-@ManyToMany), we show the annotations for the classes `DepartmentMeeting` and `Thesis` of the example of Figure 4.

```
@Entity// to make DepartmentMeeting as an entity
@Table (name="DEPARTMENTMEETING")//to reference the table
public class DepartmentMeeting{
  @Id//to make idMeeting as primary key
  @Column(name="idDepartmentMeeting")
  private int idMeeting
  ... //for the rest of attributes of the class of basic types
  @OneToMany(mappedBy="DepartmentMeeting")// to represent 1:n relations
  private Set<Thesis> thesisAnalysed;
  //getters and setters for every attributes
}
```

```
@Entity //to make Thesis as an entity
@Table (name="THESES")//to reference the table
public class ThesisProject{
  @Id//to make idThesis as primary key
  @Column(name="idThesis")
  private int idThesis
  @Column(name="sendDate")
  private Date sendDate;
  ... //for the rest of attributes of the class of basic types
  @ManyToOne// to represent n:1 relations
  @JoinColumn(name="iddepartmentmeeting")
  private DepartmentMeeting meeting;
  @OneToOne(optional=false)// to represent 1:1 relations
  @JoinColumn(name="idThesisProject")
  private ThesisProject project;
  @ManyToMany// to represent n:m relations
  @JoinTable(name="EvaluationComittee", joinColumns= {
    @JoinColumn(name="IDTHESIS")}, inverseJoinColumns = {
    @JoinColumn(name="IDDOCTOR")}
  private Set<Doctor> getEvaluationComittee;
  //getters and setters for every attributes
}
```

5 Describing the Data Object States

In order to support the third step of the methodology, we need a high level language for the business expert to describe the data object states. Also it is necessary the transformation of this description into an executable code, being possible the automatic evaluation of every objects of the database and to know the states. As it was aforementioned, it is not always possible to find the state of the data object as a string within a column of the corresponding table in the legacy database. Moreover, the state is a characteristic of the object, not mandatory of the stored data. For this reason, we represent the state as an attribute

@Transient of the mapping classes that represent business data objects. The attributes annotated with @Transient represent attributes that are part of the entity but not required to be persisted, then they do not belong to the database. When the object is loaded with the information of the database or updated by means of the setters methods, the state needs to be updated in function of the values of the remaining attributes. In Hibernate there exist call back methods that should be prefixed by annotations that describe when these methods have to be executed. Since the change of the state occurs when the object is loaded or updated, we will use the @PostUpdate and @PostLoad annotations. Both annotations represent, respectively, that the method *calculateState()* will be invoked for an entity after a constructor or update operation is executed.

The following piece of code, describes an example of how to represent the state of the Thesis data object, and the annotation of the call back method:

```
static final int CREATED = 0;
static final int COMITTEEAPPROVED = 1;
static final int ENDDEPOSIT = 2;
static final int ANALYSED = 3;
static final int EVALUATED = 4;
static final int APPROVED = 5;
static final int DENIED = 6;
static final int PRESENTED = 7;
static final int CLOSED = 8;
@Transient
private int state;
@PostLoad
@PostUpdate
public void calculateState(){
  ...
  if(...)
    state = COMITTEEAPPROVED;
  ...
}
```

5.1 Domain Specific Language for Data Object States

In order to facilitate the description by the business expert of the data object states, we propose a Domain-Specific Language (DSL) that follows the grammar presented in the left column of the following table, which is transformed automatically into the Java code of the right column. The use of the DSL permits to describe the various states of the objects, and the possible values that the objects can take to belong to them.

{ClassName}{	public class {ClassName}
	static final int {State1}=0
	...
	@PostLoad
	@PostUpdate
	private void calculateState()
#[{State1}]	if(java code transformation)
//boolean combinations of comparisons	state = {State1};
...	...
#[{StateN}]	if(java code transformation)
//boolean combinations of comparisons	state = {StateN};

With respect to the comparisons of attributes and values, some of the ones allowed by DSL are described in Table 1.

Table 1. Some DSL patterns transformation

DSL pattern	Java code pattern	Example
{attribute} IS EMPTY	get{Attribute}()==null	presentationDate is empty
{attribute} IS NOT EMPTY	get{Attribute}()!=null	presentationDate is not empty
THE NUMBER OF {atributeSet} IS {number}	get{AttributeSet}().size() =={number}	the number of evaluation comittee is 5
THE NUMBER OF {attributeSet} IS DIFFERENT OF {number}	get{AttributeSet}().size() ! ={number}	the number of evaluation comittee is different of 5
{attribute} IS EQUAL TO {value}	get{Attribute}() =={value}	Calification is equal to 'FAIL'
{attribute} IS DIFFERENT OF {value}	get{Attribute}()!={value}	Calification is different of 'PASS'
{attribute} IS LESS THAN {value}	get{Attribute}()<{value}	DepositDate is less than currentDay
{attribute} IS LESS OR EQUAL TO {value}	get{Attribute}()<={value}	DepositDate is less or equal to currentDay
{attribute} IS GREATER THAN {value}	get{Attribute}()>{value}	EvaluationDate is greater than sendDay
{attribute} IS GREATER OR EQUAL TO {value}	get{Attribute}()>={value}	EvaluationDate is greater or equal to sendDay

In Figure 5 an example of transformation between the business expert description and the Java code is presented. When an object is loaded or updated, the state of the data object will be updated, then the activity that is being executed for each data object can be known. The DSL patterns can be combined by means of boolean connects (AND, OR, NOT) in order to create a more expressive language.

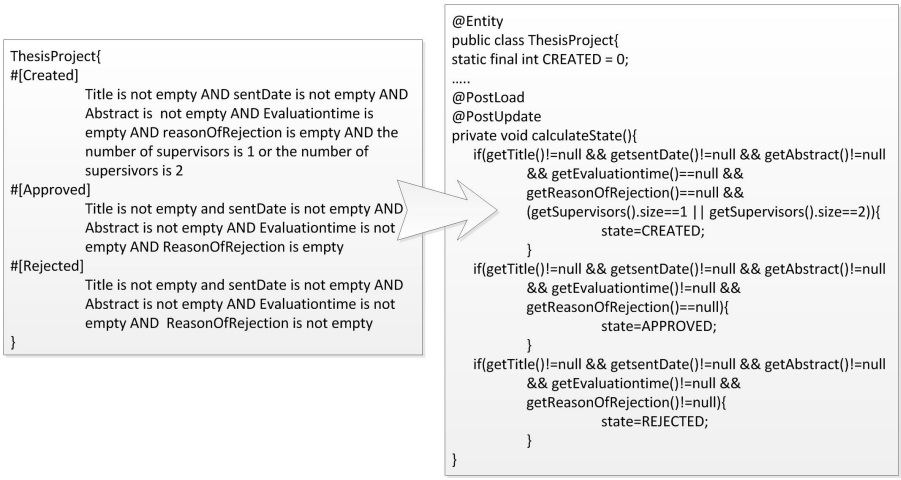


Fig. 5. Example of mapping between the DSL and Java code

6 Related Work

In the last years, several works have proposed that process and data aspects should not be examined separately. In [6], twelve business process modeling languages are evaluated with regards to their capabilities of data flow modeling against different criteria. Later, some of the previous authors have proposed an extension of BPMN data objects adding annotations to manage data dependency and instance differentiation [3], although we consider that sometimes these annotations are very low level representations (e.g. primary key and foreign key) and no significant for business stakeholders. Moreover, in this last work, the state of the data objects in the BPMN model is represented by a simple string with limited semantic value. In order to increase it, in our proposal, the states are represented by constraints or predicates over any attributes of the data object that facilitate the reasoning.

In [7] the necessity of include the data stored in the persistent layer is detected and modelled, but the solution is not object-oriented, being very difficult to be modelled for the business experts. Other works, as [8], take into account that the data are modified during the process execution, and the necessity to satisfy the activity pre-condition. The disadvantage of the paper is that the persistent layer is not taken into account, and the states are not modelled. In [9], a proposal about how to describe the object evolution during a workflow execution is presented, although the database connection with the data is missed.

On the other hand, different techniques for integrating data and control flow following the object-centric paradigm have been proposed [10][11]. A business process is modeled by its involved objects where each one has a life cycle, and multiple objects synchronize on their state changes. To derive these object life

cycles from sufficiently annotated process models, several works has been presented [12][13] in the bibliography. Although these works use data objects of the business process, but the model does not take into account, as we propose in our work, the activity-centric model that is the more used in the BPMSs. Also, the process data objects and their object life cycles could be discovered [14], but we assume that the activity-centric business process model is available, therefore no derivation is necessary.

Finally, [15] proposes a specific view for the solution of access data of business process problem: the Data Access Object Repository View that offers a fast and efficient management of DAOs. Also, the same authors have proposed a new work [16] for the data access but using Data Access Services (DAS) instead of DAOs. Although these works are very interesting for our practical choice, the objectives of these works are related to the efficiency in the data management, and not to the analysis of the data objects state model.

7 Conclusions and Future Work

In this paper, we provide the necessary support for a methodology to facilitate the migration to an activity-centric business process model. This migration implies the management of the data stored in a relational database, being necessary the data state description, and understanding the database as a repository of business data objects. We improve previous works enriching the semantic of the data object state description, using more complex objects that can involve more than one table and with n:m relations without including the relational theory in the business process model; we use all the advantages of ORM to incorporate de data objects in a more natural way into de activity-centric business process. In order to facilitate the model for the business expert, we propose a DSL to describe the data object state that is automatically transformed into Java code.

There are significant research lines that can be analysed in further depth, such as: extend the DSL to enrich the capacities to describe the data object states; determine the correctness and completeness of the data object stored in accordance with the BPM; or simulate the execution of the instances for the business process migration.

Acknowledgment. This work has been partially funded by the Ministry of Science and Technology of Spain (TIN2009-13714) and the European Regional Development Fund (ERDF/FEDER).

References

1. Weske, M.: Business Process Management: Concepts, Languages, Architectures. Springer-Verlag New York, Inc., Secaucus (2007)
2. OMG. Object Management Group, Business Process Model and Notation (BPMN) Version 2.0. OMG Standard (2011)

3. Meyer, A., Pufahl, L., Fahland, D., Weske, M.: Modeling and enacting complex data dependencies in business processes. In: Daniel, F., Wang, J., Weber, B. (eds.) BPM 2013. LNCS, vol. 8094, pp. 171–186. Springer, Heidelberg (2013)
4. Vennam, S., Dezhgosha, K.: Application development with object relational mapping framework - hibernate. In: International Conference on Internet Computing, pp. 166–169. CSREA Press (2009)
5. Elliott, J., Fowler, R., O'Brien, T.M.: Harnessing hibernate - a step-by-step guide to Java persistence. O'Reilly (2008)
6. Meyer, A., Smirnov, S., Weske, M.: Data in business processes. EMISA Forum 31(3), 5–31 (2011)
7. Gómez-López, M.T., Gasca, R.M.: Run-time monitoring and auditing for business processes data using contraints. In: International Workshop on Business Process Intelligence, BPI 2010, pp. 15–25. Springer (2010)
8. Borrego, D., Eshuis, R., Gómez-López, M.T., Gasca, R.M.: Diagnosing correctness of semantic workflow models. Data & Knowledge Engineering 87, 167–184 (2013)
9. Reggio, G., Ricca, F., Scanniello, G., Di Cerbo, F., Dodero, G.: On the comprehension of workflows modeled with a precise style: Results from a family of controlled experiments. Journal of Software and Systems Modeling (3)
10. Estañol, M., Queralt, A., Sancho, M.R., Teniente, E.: Artifact-centric business process models in UML. In: La Rosa, M., Soffer, P. (eds.) BPM Workshops 2012. LNBIP, vol. 132, pp. 292–303. Springer, Heidelberg (2013)
11. Künzle, V., Reichert, M.: Philharmonicflows: towards a framework for object-aware process management. Journal of Software Maintenance 23(4), 205–244 (2011)
12. Eshuis, R., Van Gorp, P.: Synthesizing Object Life Cycles from Business Process Models. In: Atzeni, P., Cheung, D., Ram, S. (eds.) ER 2012 Main Conference 2012. LNCS, vol. 7532, pp. 307–320. Springer, Heidelberg (2012)
13. Liu, R., Wu, F.Y., Kumaran, S.: Transforming activity-centric business process models into information-centric models for soa solutions. J. Database Manag. 21(4), 14–34 (2010)
14. Nooijen, E.H.J., van Dongen, B.F., Fahland, D.: Automatic Discovery of Data-Centric and Artifact-Centric Processes. In: La Rosa, M., Soffer, P. (eds.) BPM Workshops 2012. LNBIP, vol. 132, pp. 316–327. Springer, Heidelberg (2013)
15. Mayr, C., Zdun, U., Dustdar, S.: Model-driven integration and management of data access objects in process-driven sOAs. In: Mähönen, P., Pohl, K., Priol, T. (eds.) ServiceWave 2008. LNCS, vol. 5377, pp. 62–73. Springer, Heidelberg (2008)
16. Mayr, C., Zdun, U., Dustdar, S.: View-based model-driven architecture for enhancing maintainability of data access services. Data Knowl. Eng. 70(9), 794–819 (2011)

Change Analysis for Artifact-Centric Business Processes

Yi Wang and Ying Wang

Southwest University, Chongqing 400715, China
{echowang,waying95}@swu.edu.cn

Abstract. Business processes are subject to changes originating from customer needs and government regulations etc. A single change may set off a series of further changes which transform the initial change into flows of changes that propagate through the entire business process. This paper focuses on the challenging problem of change analysis for the three-level artifact-centric business process model. First, we identify the various types of changes that can happen to the three levels of the model. Then we present the mechanisms including sample change analysis patterns and the process element relation graph for analysing the different levels of changes. This research can reduce the complexity tasks of change management for artifact-centric business processes.

Keywords: business process management, change management, artifact-centric modeling.

1 Introduction

In today's highly dynamic and competitive environment, business processes are subject to changes and revisions originating from different customer needs, government regulations, and outsourcing partners, etc [1]. A single change in a process model may set off a series of other changes and transform the initial change into flows of changes that propagate through the entire business process. Therefore, approaches for change impact analysis and reaction are crucial for effective and efficient business process management.

The change management problem has been studied widely in the context of workflow systems [2] and service based business processes [3]. Most studies concentrate on activity-centric workflow models. In recent years, the artifact-centric business process (ACBP) modeling have attracted increasing attention [4]. In contrast to traditional workflow models which center on activity flows, the artifact-centric approaches are founded on business artifacts, which combine both data and process aspects into a holistic unit. Compared to activity-flow based approaches, the artifact-centric approach provides business managers a holistic view of the key business entities and operations. Current research for ACBPs mainly focuses on formal specification languages [5,6,7], business process specialization [8], and process discovery [9]. The challenging problem of change management for ACBPs has not been addressed in the literature.

W. Abramowicz and A. Kokkinaki (Eds.): BIS 2014, LNBIP 176, pp. 98–109, 2014.

Fig.1 shows a motivating example of the change management problems in an ACBP. We use the three-level model for ACBPs proposed in [10]. The three levels are: specification, optimization, and execution. The specification level includes **artifacts**, **abstract services** (ASs) and the associations between artifacts and ASs specified as **Event-Condition-Action** (ECA) **rules**. Each artifact has a set of attributes and a lifecycle. These declarative definitions are transformed into a (conceptual) flow diagram at the optimization level. The bottom level is the workflow system in which Web services (WSs) are used to implement individual elements of the flow diagram. Consider the following change scenarios:

(i) A change happens to the specification level, e.g., the lifecycle of an artifact needs is changed. This artifact change may have a ripple effect on the entire business process: it may require the associated ASs and the ECA rules to change. This change can also propagate to the lower levels and impact on the flow diagram and the relevant WSs. (ii) A change happens to the optimization level, e.g., an event handler is created. This change will set off a series of other types of changes at the same level. These changes will incur further changes at the bottom level, e.g., new WSs should be added for implementing the added event handler. (iii) A change happens to the execution level, e.g., a WS evolves to a new version. Then what is the impact of this change? Will it affect the elements such as artifacts at the higher levels? If some high level elements have to be modified, will these modifications propagate down to the bottom level?

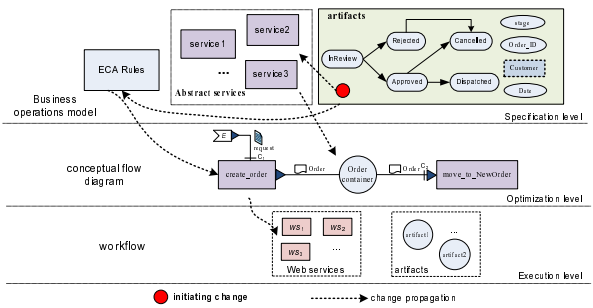


Fig. 1. A motivating example

Therefore, it is important to understand the various types of changes in an ACBP, and provide mechanisms for change analysis before solutions can be taken for coping with changes.

In this paper, we focus on the tasks of change analysis in ACBPs and make the following two major contributions. First, we present a change taxonomy based on the ACBP model defined by Bhattacharya et al [10]. Specifically, we classify the various types of changes that can happen to the three levels in an ACBP. These change types provide the foundation for the change analysis and reaction. Second, we present mechanisms for the change analysis: (i) the concept of *process element relation graph* for automatically calculating the direct impact scope of a

change; (ii) two sample change analysis patterns (CAPs) for analysing the direct impact of specification and optimization levels changes; (iii) analysis of change propagation in an ACBP.

This paper is organized as follows. In Section 2, we briefly introduce the three-level ACBP model. In Section 3, the change taxonomy is presented and Section 4 provides the change analysis mechanisms. Section 5 is the related work and Section 6 concludes this paper.

2 The Artifact-Centric Business Process Model

Before we start presenting the change analysis mechanisms, we briefly introduce the three-level ACBP model proposed in [10]. The model consists of three levels: specification, optimization and execution.

2.1 The Specification Level

The specification level defines data (artifacts) and the business process (ASs and ECA rules) in a declarative manner. Artifacts describe key business entities, for example, a purchase order, a customer, and a payment in a sales business process. Each artifact has an information model (c.f. Fig.2(a)). The Order artifact includes: stage, orderID, productList, and date. Artifacts can refer to each other. Here, a Customer artifact (depicted as a dashed rectangle) relates to the Order. The lifecycle of an artifact is described by a finite-state machine. Each state of the machine corresponds to a stage in the lifecycle of the artifact. As shown in Fig.2(b), five stages are identified in the lifecycle of an Order artifact.

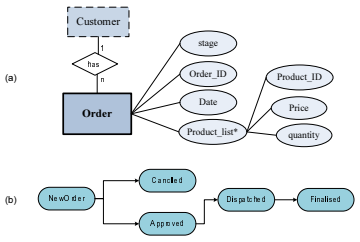


Fig. 2. The Order artifact. (a) Information model; (b) lifecycle

An AS is defined by: inputs, outputs, preconditions, and effects [11]. The inputs and outputs of an AS can be artifacts and attributes. The preconditions specify a set of required conditions for invoking an AS. Effects specify what the possible results an AS will have. ASs make changes to artifacts, e.g., evaluating an attribute or moving an artifact to a new stage. Each AS is linked to a primary artifact. For example, the ASs create_order and approve_order are associated to the Order.

How ASs are associated to artifacts are specified as a set of ECA rules. We provide an example ECA rule:

R1 Create Order.

Event: a request from customer C to create an Order instance for products list [product_id, price, quantity].

Condition: C is registered, the products in the list are available, and the quantity for each product is below the limit.

Action: invoke create_order(C, list [product_id, price, quantity]).

2.2 The Optimization Level

The optimization level contains the (**conceptual**) **flow diagram** which is a procedural representation of the specification level. The flow diagram conforms to the high level design and hiding implementation details. Optimization of the workflow can be conducted at this level. Fig.3 shows part of a flow diagram. A rectangle denotes an AS. Artifacts are stored in containers. The arrows denote information flows. Once a request is received from a customer, and the condition C_1 is satisfied, the create_order service is invoked and an Order instance is generated. Then, the move_to_NewOrder service is invoked under condition C_2.

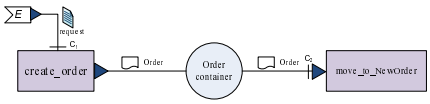


Fig. 3. The (conceptual) flow diagram

2.3 The Execution Level

This level contains the workflow system in which WSs are used to implement individual components of the flow diagram. For example, an AS may be implemented by an elementary WS or a composite WS. A WS may belong to the organization (e.g., the WS for moving an artifact to a new stage) or its partner (e.g., a payment service). It may also be selected at runtime dynamically from the Web.

3 The Change Taxonomy

Based on the three-level ACBP model, we present the change taxonomy (Fig.4) as the foundation for change impact analysis.

3.1 Specification Level Change

The specification level contains artifacts, ASs, and ECA rules. Thus, we classify the major types of changes at this level into artifact change, AS change, and ECA rule change. The detailed classification is shown in Fig.4.

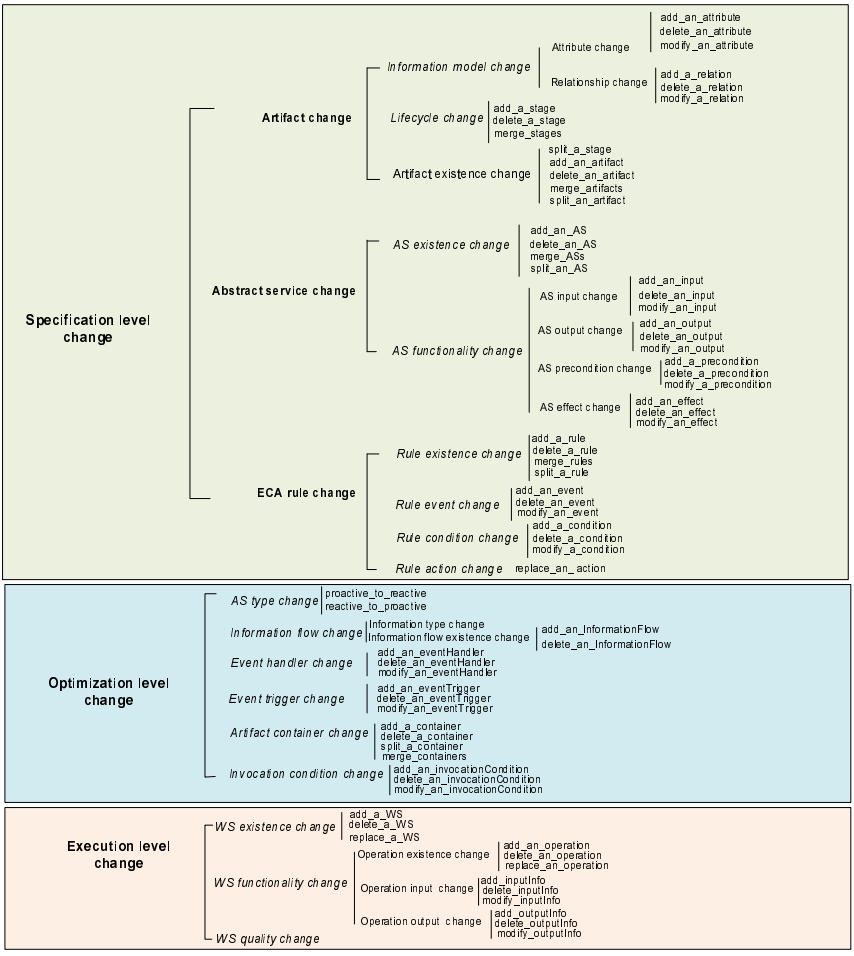


Fig. 4. The change taxonomy

A. Artifact change includes: (1) *information model change* has two types of sub changes: attribute change and relationship change. Attribute change refers to modifications on the data structure of artifacts. Relationship change is about modifying the associations among different artifacts. (2) *Lifecycle change* refers to modifications on the lifecycle of an artifact. A new stage can be inserted to the lifecycle. An existing stage may be removed. Multiple stages can be merged and a single stage can be split into several stages. (3) *Artifact existence change* refers to adding a new artifact, removing an existing artifact and merging artifacts or splitting an artifact.

B. AS change includes (1) *AS existence change* and (2) *AS functionality change*. The AS functionality change is further categorized into AS input change, AS output change, AS precondition change, and AS effect change.

C. ECA rule change includes (1) *rule existence change*, (2) *rule event change*, (3) *rule condition change*, and (4) *rule action change*.

3.2 Optimization Level Change

The optimization level contains the (conceptual) flow diagram. Elements in a flow diagram are event handlers, ASs, information flows, invocation conditions, and artifact containers. We identify the following types of changes that can happen to a flow diagram. A. *ASs type change*. An AS can be reactive or proactive. A proactive AS is running all the time and does not need to be invoked. On the contrary, a reactive AS needs to be invoked. A proactive AS can change to a reactive one (proactive_to_reactive) and vice versa (reactive_to_proactive). B. *Information flow change* includes the information type (an artifact or a message) change and information existence change. C. *Event handler change* refers to modifications on event handlers in a flow diagram. For example, a new event handler can be created or modified. D. *Event trigger change* refers to modifications on event triggers in a flow diagram. For example, a new event trigger can be created or modified. E. *Artifact container change* refers to modifying the container design of a specific artifact. For example, an existing artifact container can be split into several containers. F. *Invocation condition change*. A condition in a flow diagram corresponds to the condition in an ECA rule at the specification level. A condition change includes modifying a condition, adding a new condition and removing an existing condition.

3.3 Execution Level Change

The execution level contains WSs that implement individual elements in the flow diagram. WSs are subject to constant changes, e.g., WS evolution. A WS is defined by a tuple: $ws = (M, O)$, where M=InMsg$\cup$OutMsg contains the input and output messages, O is the set of operations. The notations OutInfo (op) and InInfo (op) refer to the output and input basic data types of operation *op* extracting from its input and output messages [12]. The WS change can be categorized into: A. WS existence change, B. WS functionality change, which is further classified into: *operation existence change*, *operation input change* and *operation output change*, and C. WS quality change.

4 Change Impact Analysis

In this section, we discuss how to analyse the impact of a specific change. The change analysis process is given in Fig.5. First, an initiating change is made; usually a number of other changes need to be performed in order to support the initiating change. For instance, to make an attribute change (modify_an_attribute) may require AS changes (e.g., modify the inputs or outputs of an existing AS).

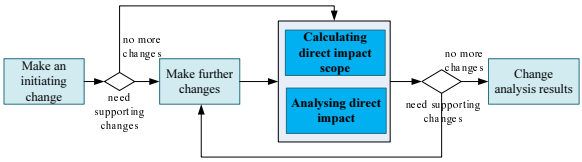


Fig. 5. The change analysis process

After the initiating change and its supporting changes are made, the direct impact is obtained. If further changes are required, the direct impact will be analysed until no more changes are made. Finally the entire change analysis results are presented.

The direct impact and scope of a change refers to the process elements that are directly affected by the change. The term *directly* here means that the effect of a change is determined by the dependencies between process elements. For example, if we modify an ECA rule at the specification level, then the corresponding information flows in the flow diagram will need to be changed. A single change may set off a series of other changes. Thus, an initiating change may cause flows of changes that propagating through the entire process. We call this type of change impact as the ripple effect. In the following, we shall first discuss how to analyse the direct impact of changes at the three levels respectively. Then we discuss how the change propagation can be analysed based on the direct impact analysis results.

4.1 Calculating Direct Change Impact Scope

The direct impact scope of a specific change refers to the process elements that are affected by the change directly. The calculation is based on the associations between the elements. To facilitate automatically computing the direct impact scope of a change, we propose the concept of *process element relation graph*. Fig.6 shows part of an example process element relation graph. The nodes in the graph are process elements, e.g., an attribute, an AS and its input/output parameters. A directed edge with a label specifies the relationship between two elements. In Fig.6, e.g., the hasAttribute label on the edge specifies that the artifact A has an attribute $attri_1$. The AS s is linked with implementedBy to a set of WSs (ws_1 and ws_2) which have similar functionality. Here, the hasAttribute relationship can be obtained from the information model of artifacts and the implementedBy relationship can be established based on the realisation of each AS. In a word, the process element relation graph can be generated from the definition of the ACBP. Based on the graph, when a process element is changed, we can locate its direct impact scope by including all the nodes that has a link to and from the element. E.g., if s is modified and we find the related links are: (s, hasParameter, $attri_2$), (s, implementedBy, ws_1), (s, implementedBy, ws_2), (rule1, hasAction, s), (A hasService, s), then the impact scope contains:$attri_2, ws_1, ws_2, rule1, A$.

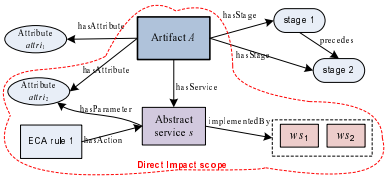


Fig. 6. The process element relation graph

4.2 Analysing Direct Change Impact

We discuss how to analyse the direct impact of a specific change on the ACBP model in this subsection.

Analysing Direct Impact of Specification Level Change. In order to understand what kind of impact a specification change can have, we design a set of change analysis patterns (CAPs) to capture the knowledge about the effects of the each bottom change types in the taxonomy (Fig.4). The approach of using patterns have been used for the change management of workflow systems [2], WS compositions [13,3], and software development processes [14]. Pattern based approach is effective because patterns characterize generalized knowledge in the change management and can be reused and extended in different context. Based on the dependencies between the elements in an ACBP, we develop the CAPs with the aim to support the automation of change management. Each CAP describes the **possible** *direct* effect of a specific type of change. A CAP contains a pattern name, the change and its possibly caused impact on the process elements represented as change types, and a text description. Owing to space limitations, we show an example CAP: modify_an_attribute (Fig.7). The dashed arrows depicts the possible impact of the change. Different arrows have the "OR" relationship. This pattern describes that if an artifact attribute is modified, the related process elements (e.g. an AS input) might be affected. Therefore, the possible effect caused by this type of change is: AS functionality change, rule event change, and/or rule condition change. The reason for having these effects is that an attribute can appear in the inputs/outputs, the precondition/effects of ASs, and the event/condition of ECA rules. Thus when it is modified, these related process elements might be affected and modified accordingly. Before the change really happens, we can not know the exact impact. But by incorporating the direct impact scope calculated by the process element relation graph and the corresponding CAP, we can have a more accurate understanding of the impact for a change.

Analysing Direct Impact of Optimization Level Change. The optimization level contains the flow diagram, which is constructed based on the declarative specification of artifacts and ASs and is a procedural presentation of the business process. The individual elements (e.g. event handlers) in the flow diagram are implemented by WSs at the execution level. Thus, an optimization

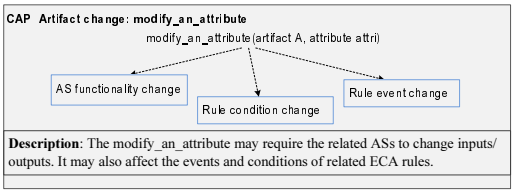


Fig. 7. The CAP for modify_an_attribute change

level change usually requires the corresponding implementation to be modified accordingly. Similar to the analysis for specification level changes, we design the CAPs for capturing the effects of optimization level changes based on the dependencies between process elements. The CAP in Fig.8 is the possible effect of the add_an_eventHandler change. If an event handler is created for an AS, the associated information flows between the event handler and the AS need to be created for transmitting the corresponding data. In addition, a WS needs to implement the new event handler. Thus, a WS existence change (adding a new WS for realising the event handler) or a WS functionality change (modifying an existing WS for handling the new event) may happen.

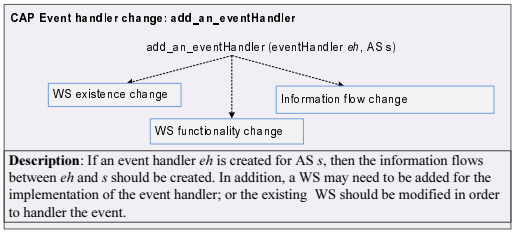


Fig. 8. The CAP for add_an_eventHandler change

Analysing Direct Impact of Execution Level Change. The execution level contains WSs that implement individual elements of the flow diagram. A WS change may originate from an internal or external origin. Internal origins include specification level changes such as artifact and abstract service changes. The high level changes usually require the corresponding WSs to change accordingly. An external origin originating from the environment can be WS evolution. In the following, we focus on the analysis for WS functionality change.

A. Operation Existence Change. If an operation is deleted from a WS, the element in the flow diagram that relies on the operation will be affected and this change causes significant problems to the workflow system. If a new operation is added to a WS, this change normally will not impact on the system.

B. Operation Input and Output Change (i). OutInfo(op')<OutInfo(op): the operation op changes by deleting its output data types. This change suggests that the corresponding element in the flow diagram that uses the operation may

be affected. Suppose the AS s relies on op. We use OutInfo(s) to represent the information in the output parameters of the AS s. First we should know whether the information OutInfo (op)-OutInfo(op') is used by s or not. If the former is bigger than the latter, the implementation of s is affected by this change. If not, the change has not impact on s. The input data types of op can also change. If InInfo (op')$\leq$ InInfo (op), it means that op requires less input data. This change will impact on the realization of s. If InInfo (op') $>$ InInfo (op), then the change may cause the workflow system fail to invoke the operation op. In this case, we need to check if the information difference InInfo(op')- InInfo(op) can be provided by the associated artifacts of s or not. **(ii)** OutInfo(op')$>$ OutInfo(op): this change suggests that the functionalities of the element in the flow diagram can be realized by op'. If InInfo (op')$\leq$ InInfo (op), we know that this change requires no further changes in the workflow system. If InInfo (op') $>$ InInfo (op), we need to check if the corresponding artifacts can provide the data: InInfo (op') - InInfo (op).

4.3 Change Propagation

The above change analysis focuses on understanding the *direct* effect of a change. As process elements are connected to each other, a single change normally will propagate to the related process elements at different levels. That is, an initiating change usually sets off a number of further changes and thus flows of changes occur. The change propagation analysis help understand the ripple effect of a change. The propagation process can be obtained based on the direct impact analysis. Fig.9 provides an illustrating example. First the modify_an_attribute change is made. Suppose the initiating change triggers an AS functionality change: modify_an_input and a rule condition change: modify_a_condition. For the modify_an_input change, the associated WS functionality may need to change. For the modify_a_condition change, the corresponding invocation condition in the flow diagram will need to be modified. The dashed arrows depict the flows of changes triggered by the initiating change. To sum up, the ripple effect of an initiating change can be obtained by using the direct impact analysis mechanisms for each triggered change during the change propagation process (see Fig.5).

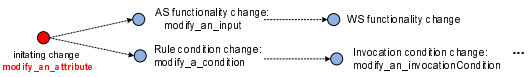


Fig. 9. Change propagation

5 Related Work

The change management problem has been studied widely for workflow systems [3]. Most studies concentrated on process-centric workflow models. Weber, Reichert, and Rinderle-Ma [2] present patterns and support features for structure

changes in process models, which facilitate the comparison of existing change frameworks simpler. Changes in the context of a cross-organizational setting are studied in [16]. In [17], the authors study the change propagation from a centralized process to its derived decentralized partitions. Gerth and Luckey [18] introduce a framework for changes in business process models. The framework consists of seven components: abstraction to the intermediate representation, matching, difference detection, dependency analysis, equivalence and conflict analysis, and merging.

Artifact-centric approaches to business process modeling have been studied in a number of works [5,7,6,8,9]. Gerede and Su [5] develop a specification language based on computation tree logic to specify artifact behaviors for artifact-centric business process models. Sun et al [6] propose a declarative choreography language for artifacts in both type and instance levels. Based on Petri nets, Lohmann and Wolf [7] extend existing models with agents and locations. Business processes can access artifacts from their locations with the help of agents. In [8], the process specialization for artifact-centric process models is studied. The authors provide formal analysis of behavioral consistency between a specialized process and its base process. Nooijen et al [9] present an automatic technique for artifact-centric business processes discovery given a relational database that stores process execution information of a data-centric system.

Our work is different from the above research because we focus on the change analysis problem in three-leveled ACBPs where different levels of changes may occur and cause various type of effects.

6 Conclusion

In this paper we have provided an approach to dealing with the change management in the three-level ACBP model which focuses on business artifacts. The taxonomy for changes associated with the specification, optimization, and execution levels in an ACBP is presented. Based on the identified various types of changes, we provide the mechanisms for change impact analysis, including: the process element relation graph for calculating the direct impact scope, a set of CAPs for analysing the direct change effect, and the process of change propagation. These mechanisms allow for developing reactions for handling the various types of changes and make contributions for change automation in ACBPs. Our future work includes the development of a change management tool and identifying co-change relations in ACBPs.

Acknowledgements. This research is sponsored by the National Natural Science Foundation of China (61303229), PhD Research Programs Foundation of Southwest University (SWU112030), and Scientific Foundation for the Returned Overseas Chinese Scholars, State Education Ministry.

References

1. Papazoglou, M.P.: The Challenges of Service Evolution. In: Bellahsène, Z., Léonard, M. (eds.) CAiSE 2008. LNCS, vol. 5074, pp. 1–15. Springer, Heidelberg (2008)
2. Weber, B., Reichert, M.R., Rinderle-Ma, S.: Change Patterns and Change Support Features - Enhancing Flexibility in Process- aware Information Systems. Data Knowl. Eng. 66(3), 438–466 (2008)
3. Wang, Y., Wang, Y.: A Survey of Change Management in Service-Based Environments. Service Oriented Computing and Applications 7(4), 259–273 (2013)
4. Hull, R.: Artifact-Centric Business Process Models: Brief Survey of Research Results and Challenges. In: Meersman, R., Tari, Z. (eds.) OTM 2008, Part II. LNCS, vol. 5332, pp. 1152–1163. Springer, Heidelberg (2008)
5. Gerede, C.E., Su, J.: Specification and Verification of Artifact Behaviors in Business Process Models. In: Krämer, B.J., Lin, K.-J., Narasimhan, P. (eds.) ICSOC 2007. LNCS, vol. 4749, pp. 181–192. Springer, Heidelberg (2007)
6. Sun, Y., Xu, W., Su, J.: Declarative Choreographies for Artifacts. In: Liu, C., Ludwig, H., Toumani, F., Yu, Q. (eds.) Service Oriented Computing. LNCS, vol. 7636, pp. 420–434. Springer, Heidelberg (2012)
7. Lohmann, N., Wolf, K.: Artifact-Centric Choreographies. In: Maglio, P.P., Weske, M., Yang, J., Fantinato, M. (eds.) ICSOC 2010. LNCS, vol. 6470, pp. 32–46. Springer, Heidelberg (2010)
8. Yongchareon, S., Liu, C., Zhao, X.: A Framework for Behavior-Consistent Specialization of Artifact-Centric Business Processes. In: Barros, A., Gal, A., Kindler, E. (eds.) BPM 2012. LNCS, vol. 7481, pp. 285–301. Springer, Heidelberg (2012)
9. Nooijen, E.H.J., van Dongen, B.F., Fahland, D.: Automatic Discovery of Data-Centric and Artifact-Centric Processes. In: La Rosa, M., Soffer, P. (eds.) BPM Workshops 2012. LNBIP, vol. 132, pp. 316–327. Springer, Heidelberg (2013)
10. Bhattacharya, K., Hull, R., Su, J.: A Data-Centric Design Methodology for Business Processes. In: Handbook of Research on Business Process Modeling, pp. 503–531 (2009)
11. Martin, D., Burstein, M., Hobbs, J., et al.: OWL-S: Semantic Markup for Web Services, 22: 2007-04. W3C member submission (2004)
12. Fokaefs, M., Mikhaiel, R., Tsantalis, N., et al.: An Empirical Study on Web Service Evolution. In: 2011 IEEE International Conference on Web Services, pp. 49–56. IEEE (2011)
13. Benatallah, B., Casati, F., Grigori, D., Nezhad, H.R.M., Toumani, F.: Developing Adapters for Web Services Integration. In: Pastor, Ó., Falcão e Cunha, J. (eds.) CAiSE 2005. LNCS, vol. 3520, pp. 415–429. Springer, Heidelberg (2005)
14. Maurer, F., Dellen, B., Bendeck, F., et al.: Merging project planning and Web enabled dynamic workflow technologies. IEEE Internet Computing 4(3), 65–74 (2000)
15. Rinderle, S., Wombacher, A., Reichert, M.: Evolution of Process Choreographies in DYCHOR. In: Meersman, R., Tari, Z. (eds.) OTM 2006. LNCS, vol. 4275, pp. 273–290. Springer, Heidelberg (2006)
16. Wombacher, A.: Alignment of Choreography Changes in BPEL Processes. In: International Conferenceon Services Computing, pp. 1–8 (2009)
17. Fdhila, W., Rinderle-Ma, S., Baouab, A., et al.: On Evolving Partitioned Web Service orchestrations. In: 5th IEEE International Conference on Service-Oriented Computing and Applications, pp. 1–6. IEEE (2012)
18. Gerth, C., Luckey, M.: Towards Rich Change Management for Business Process Models. Softwaretechnik-Trends 32(4), 32–34 (2012)

Component-Based Development of a Metadata Data-Dictionary

Frank Kramer and Bernhard Thalheim

Christian-Albrechts-University Kiel, Computer Science Institute
24098 Kiel, Germany

Abstract. Metadata provide information about data, ease access and querying, support well-planned evolution, and allow to reason on data quality. Moreover, heterogeneous data sources are better to integrate if well-defined metadata are available. Therefore, metadata management becomes a crucial element of modern database systems. We develop a component-approach to metadata management. This approach generalises classical data dictionaries and is based on many-dimensional schemata each dimension of it represents a specific facet of the schema.

Keywords: Conceptual Modeling, Metadata, Component-based Development, Metadata Management, Metadata Repository.

1 Introduction

An important part in the field of data management is the metadata management. Metadata support heterogeneous user groups to get information and assessment over the corresponding data within a system. The management of metadata becomes complicated, if there is a great variability of databases that covers the metadata. Furthermore, there is an evolution in the metadata and the underlying schema over time. Thus, a system is needed that allows a dedicated management of the metadata and that is also safe against evolution of the metadata. Today there exist different approaches to metadata management systems. They can be divided into generic or high specialized approaches. The generic approaches give only suggestions for what is needed for good metadata management systems [1]. Specialized approaches are constructed for special applications such as data warehousing [2] or master data management [3]. Two movements of metadata management can be found often that cover both their own problems. On the one hand, the metadata are integrated into the global schema. Such a schema can become quickly confusing, unreadable and not extensible. On the other hand, the metadata can be disembodied from the application data and worfklow data in an external repository. This leads to a high effort for tending, reading and connecting the metadata with other data within a special context.

Functional approaches for solving problems in large database applications can be found within the field of the *Conceptual Modelling in the Large* (CoMoL). It describes special techniques to model large applications. This covers the structure, functionality, interaction and support for such applications. All described

W. Abramowicz and A. Kokkinaki (Eds.): BIS 2014, LNBIP 176, pp. 110–121, 2014.

techniques exceed the classical techniques for modelling smaller applications. One facet of CoMol comprises the component-based development of information systems. A component is *a database schema that has an import and an export interface by which it may be connected to other components via a standardized interface technique* [4]. With these components we get a modularization of a schema with all the advantages like loose coupling and abstraction from components on their content. With these components we are able to scale a global schema into a metadata, workflow data and application data dimension. As a result, we reduce the complexity of the whole schema drastically.

This paper presents an approach to metadata management that is based on an internal metadata Data-Dictionary. The dictionary is a generalization of a database data-dictionary. To construct the metadata data-dictionary we define six disjoint categories that can be realized as components. The components build a metadata dimension. The connection between the dimensions is enabled with a concept called harnesses. These harnesses build the metadata data dictionary that corresponds to a metadata repository. As distinct from other repositories, the metadata data-dictionary is not disembodied from the system. Section 2 will give a closer look at creating components. Furthermore, we demonstrate how we can scale data into dimensions based on this components and how the dimensions can be connected together with harnesses. Then, the section 3 will show, how a generalized metadata data-dictionary can be realized. Therefore, we define the six disjoint metadata categories and apply the component-based development on these categories. As a result, we get the metadata data-dictionary. We will show also that such a dictionary meets all requirements of a metadata repository. After that, section 4 will give a short look on related work in the field of metadata management. Finally, a conclusion and a short outlook on future research will be given in section 5.

2 Component-Based Development

In this section, we present a component-based development of database schema. Therefore, we will describe database components as presented in [5] and [4]. After introducing the components, we want to demonstrate how they can be used to dimension a global schema and how the dimensions can be connected together.

2.1 Components in a Nutshell

For our definition of a database component we use the Higher-Order Entity-Relationship Model (HERM) [6]. In HERM a database type is defined as $\mathfrak{S} = (\text{Struc}, \text{Op}, \Sigma)$ with a structure Struc, a set of operations Op and a set of static integrity constraints Σ. The structure is defined by a recursive type equality $t = B|t \times \ldots \times t\ |[t]|\{t\}|l{:}t$ over a set of basic data types B, a set of labels L and constructors for tuple (product), set and bag. A database schema $S = (\ \mathfrak{S}_1, \ldots, \mathfrak{S}_m, \Sigma_G\ )$ is given by a set of database types $\mathfrak{S}_1, \ldots, \mathfrak{S}_m$ and a set of global integrity constraints Σ_G.

Formally, a component can be described as input-output machines. Every machine gets a set of all database states S^C, a set of input views $I^{\mathfrak{V}}$ and a set of output views $O^{\mathfrak{V}}$. A view can be defined as $\mathfrak{V} = (\mathrm{V}, Op_V)$ with an algebraic expression V on a database schema S and a set of HERM algebra operations Op_V on the view V. The views are used for the collaboration of the components by exchanging data over them. Therefore, an input view of one machine can be connected to an output view of another machine. This data exchange is done by a channel C. The structure of the *channel* is defined by a function $type : C \to \mathfrak{V}$ that maps a channel C on a corresponding view schema V. In general, the input and output sets from a component can be seen as words from a set of words M^* of the underlying database structure.

Thus, a database component is defined as $\mathfrak{K} = (S_{\mathfrak{K}}, I^{\mathfrak{V}}_{\mathfrak{K}}, O^{\mathfrak{V}}_{\mathfrak{K}}, S^C_{\mathfrak{K}}, \Delta_{\mathfrak{K}})$ with a database schema $S_{\mathfrak{K}}$ that describes the database schema of $\mathfrak{K}$, a syntactic interface composed of a set of input views $I^{\mathfrak{V}}_{\mathfrak{K}}$ and output views $O^{\mathfrak{V}}_{\mathfrak{K}}$, a set of all database states $S^C_{\mathfrak{K}}$ and a channel function $\Delta_{\mathfrak{K}} : (S^C_{\mathfrak{K}} \times (O^{\mathfrak{V}}_{\mathfrak{K}} \to M^*)) \to \mathfrak{P}(S^C_{\mathfrak{K}} \times (I^{\mathfrak{V}}_{\mathfrak{K}} \to M^*))$ that connects an output view with a set of input views. To connect two components together they must be free of name conflicts and the input and output views have to be domain-compatible. Assume two components $\mathfrak{K}_1 = (S_1, I^{\mathfrak{V}}_1, O^{\mathfrak{V}}_1, S^C_1, \Delta_1)$ and $\mathfrak{K}_2 = (S_2, I^{\mathfrak{V}}_2, O^{\mathfrak{V}}_2, S^C_2, \Delta_2)$. They are free of name conflicts if the names of their entity, relationship and attribute names within their schema S_1 and S_2 are disjoint. Two channels C_1 from $\mathfrak{K}_1$ and C_2 from $\mathfrak{K}_2$ are domain-compatible, if $dom(type(C_1)) = dom(type(C_2))$. So the output $O^V_1 \in O^{\mathfrak{V}}_1$ of component $\mathfrak{K}_1$ is domain-compatible to input $I^V_2 \in I^{\mathfrak{V}}_2$ of component $\mathfrak{K}_2$ when $dom(type(O^V_1)) \subseteq dom(type(I^V_2))$. For the definition of unification, permutation and renaming of channels together with the introduction of fictitious channels and the parallel composition of channels with feedback we refer back to [4].

2.2 Dimensioning of Data

The modularisation of a database schema with components is then used to scale the schema into dimensions. To do this, the assumption is made that the whole schema is transformed into a component-based schema. After this transformation the data are partitioned into three orthogonal dimensions for application, workflow and metadata, as done in [7]. The application data tier contains all the data that are used by the application. For example, measured data from a research cruise could be seen as application data. The workflow tier contains all data about the structure and process of the workflows that exist within the application. The procedure steps of taking soil samples from the ocean can be such workflow data. The metadata tier contains all metadata that exist within the system. Longitude and latitude of a soil sample can be stored in this tier. To connect the dimensions, we will use a concept, we called harness because the behaviour is similar to wired harnesses in electrical engineering. A detailed description of harnesses is given in section 2.3. Figure 1 shows how the dimensioning can look like.

With the dimensioning of a schema, it is possible to store data based on their origin and purpose. This reduces drastically the complexity of the whole

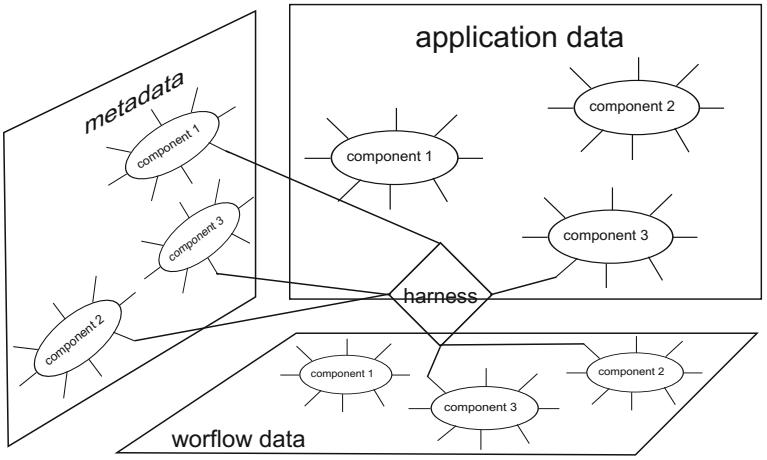


Fig. 1. Data Dimensions

database schema. Because of the dimensioning the data there exist only small components and not a huge global schema. Problems such as saving the same data on different parts of the schema are avoided. Every component can be connected over harnesses to all schema parts where it is needed.

2.3 Harnesses

In this section, we want to present the concept of harnesses. They are based on the work of [4]. A harness is based on a harness skeleton. This is a special form of metaschema architecture. The skeleton consists of a set of components and a set of harnesses that represent the overlapping functions for the components. A n-ary harness skeleton can be defined as a triple $\mathfrak{H} = (\mathcal{K}, \mathcal{L}, \tau)$ with a set of components $\mathcal{K} = \{\mathfrak{K}_1, \ldots, \mathfrak{K}_m\}$, a set of lables $\mathcal{L} = \{L_1, \ldots, L_n\}$ having $n \geq m$ that represent roles of components in the skeleton and a total function $\tau : \mathcal{L} \to \mathfrak{K}$ that assigns a component to its roles. Figure 2 displays an example of a graphical representation of a harness skeleton.

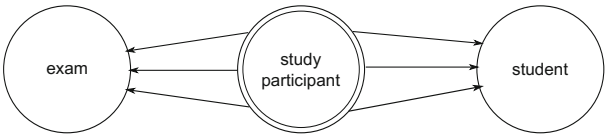


Fig. 2. Graphical harness skeleton

A student and an exam are components. They are represented as single edge circles. The double edge circle represents a harness skeleton that connects the

student and the exam to a study participant. Components can be connected to other components in a great variety. In the classical schema design this will lead to a huge and confusing schema because for every introduced subschema a new usage type must be introduced too. To avoid this problem for the component-based design, a filter can be defined for the skeleton. A filter connects the views of the components with the labels of the harness skeleton. Let $\mathfrak{H} = (\mathcal{K}, \mathcal{L}, \tau)$ be a n-ary harness skeleton having m components $\mathfrak{K}_j = (S_j, I_j^{\mathfrak{V}}, O_j^{\mathfrak{V}}, S_j^C, \Delta_j)$ with $1 \leq j \leq m$. Furthermore, let $V_1^{\mathfrak{K}_j}, \ldots V_{l_{\mathfrak{K}_j}}^{\mathfrak{K}_j}$ be all input and output views of a component $\mathfrak{K}_j$. A filter $\mathcal{F} = (\mathcal{L}, \iota)$ connects a view from the component $\mathfrak{K}_j$ with a label L_i, if $\iota(L_i) = l$ for $j = \tau(L_i)$ and $l \in \{V_1^{\mathfrak{K}_j}, \ldots V_{l_{\mathfrak{K}_j}}^{\mathfrak{K}_j}\}$ holds. A harness is then defined as $\mathcal{H} = (\mathcal{K}, \mathcal{L}, \iota \circ \tau)$ composed of a harness skeleton $(\mathcal{K}, \mathcal{L}, \tau)$ and a filter $(\mathcal{L}, \iota)$. With this harness we are able to connect components in different dimensions of a dimensioned schema. After the formal construction of components and harnesses for dimensioning data, we want to use these constructs to create our approach of metadata management based on components.

3 Metadata Management

Metadata management covers all functions a system must provide for creation, maintenance and deployment of metadata [2]. Such a management system is normally represented by an independent repository system that is separate from the other system. In this section, we will present a metadata repository that covers all the provided functions of a management system without separating it. Beforehand, we present our approach of such a repository, and we take a quick glance at what kind of data is defined as metadata.

Generally, metadata are defined as *data about data* [1],[8],[3],[2]. In this paper, we want to use a more specialized definition from [1]. Therefore, we first introduce instance data. *Instance data* cover all data that are used as input into a tool, an application, a database or other process engines. All data that describe the format and characteristic of instance data are called *metadata*.

One problem of separating data into instance data and metadata is the fact that the separation depends on the user's point of view on the data. Take as example a relation person = (EMail, familyName, firstName, address, fon). If a system developer uses this relation the data within the rows are the instance data he wants to work with. The caption of the columns is the metadata for him because they describe what instance data the developer can find in the columns of a row. If an administrator of the database looks on the same relation, the caption of the columns is the instance data he works with. The data within the rows are not of interest for him. Information about the domain of the columns and the disc space usage are then the metadata for his instance data. As a consequence, instance data and metadata can not be directly separated. They are always in relation to the user's point of view and the context of the user. Thus, in the next step we have to structure metadata in a global context to avoid the user's point of view on the data [2],[1].

3.1 Metadata Categories

In this section, we categorize metadata in a global context to describe the different issues they are used for. For this categorization, we are only interested in functional metadata [9], [2]. These are metadata that are used to interpret application data and to recognize correlations between the data. Technical metadata are not dealt with in this paper as they describe the structure and the behaviour of application data. In general, technical metadata can be found in all modern systems as, for example, the data-dictionary of a database management system.

To divide the functional metadata into categories, we use the fact that metadata are used to describe instance data. This can be represented by the classical W6H-questions: *who*, *what*, *when*, *where*, *why*, *how* and *by what means*. These questions are first mentioned in the classical rhetoric of Hermagoras of Temnos[1] and can be found in the *Zachman Enterprise Architecture Framework* for Information Systems [10]. Therefore, we will use only a single metadata object to answer one of these questions. In the next step, we generate categories using an extended W6H-question set. Every category covers the questions that belong together and that are disjoint to the other questions. After this, we get a set of metadata categories that are shown in Figure 3.

As a result, we get six disjoint categories that cover a set of metadata. Because of answering exactly one question, every metadata can be classified exactly into one of these categories. *Time and Space* covers all relevant metadata about the time and the space for the application data. The *Quality* component contains the metadata for quality information. *Service* metadata gives information about the reason why data is in the system and the method how the data comes into the system. The *Administrative* category covers the information about all administrative information such as rights on the data or roles of the user. The *Structure* consists of metadata that describe the structure of the data it belongs to. *Provenance* covers the history of data from the creation to the deletion of data in the system.

Thus, we have found a way to divide metadata into six global categories that are independent from the user's point of view. Every functional metadata in a system could be assigned to one of these categories. Therefore, a model is needed that can map these categories on an application view. This will build a metamodel as defined in [1] for our categories. For this, we use the component-based development of database schemes from section 2.1. Consequently, every metadata category becomes a single component in the metadata dimension. The advantage of taking the component based approach is that we get all advantages from this approach for every metadata category, for example, the easy extension of a schema and the good understandable schema of the component. The last part missing from our approach is the connection between the metadata components and the corresponding application data components.

[1] Unfortunately, the work of Hermagoras of Temnos is lost. However, the Roman author Cicero refers back to the work of Temnos in his opus "'*De Inventione*"'. Therefore, parts of Temnos' work are still available to the present day.

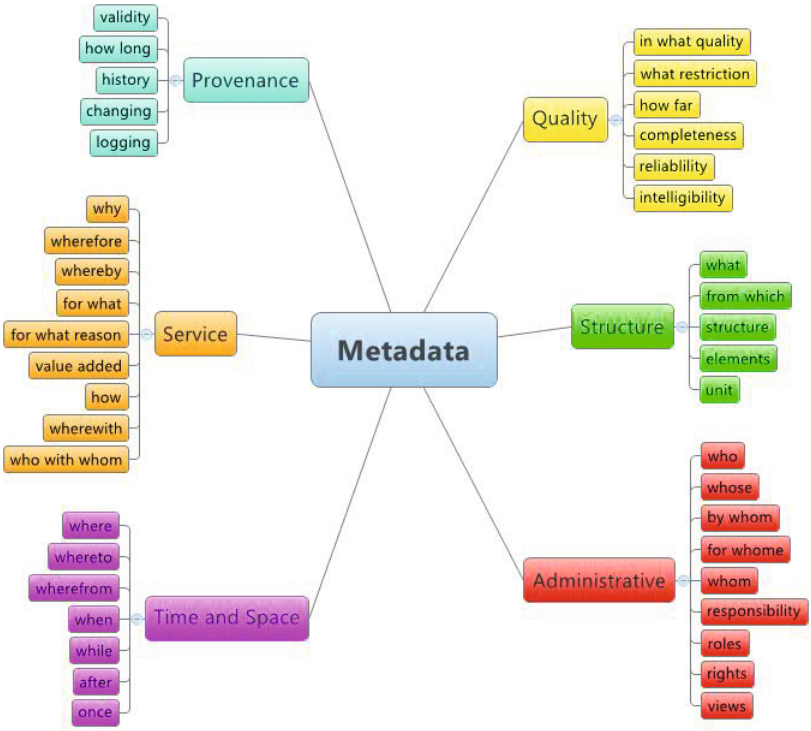


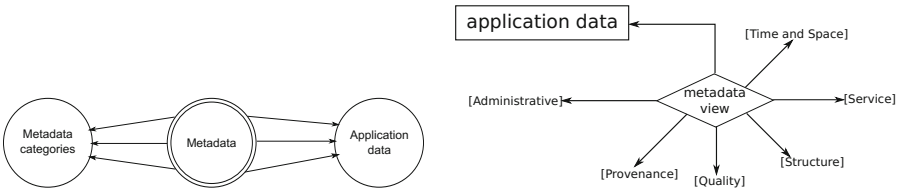
Fig. 3. Metadata categories

3.2 Metadata Data-Dictionary as Repository

The last step for our metadata management approach is the connection between the metadata components to the components in the other dimensions. In this section, we will only describe the connection of metadata with application data. The connection to the workflow data can be realized in the same way. In 2.3 we have described the concept of harnesses to connect components in different dimensions. We will use these harnesses to connect the metadata categories with the application data. These harnesses are then realized as a metadata data-dictionary that represents an internal metadata repository for a system.

Data-Dictionaries can be found primarily in the area of database management systems (DBMS). They represent a set of system tables that covers different information like the definition of schema objects, logging from actions, comments for tables and rows and much more. For example, take the SYSIBM table from the DBMS DB2. They cover information about the structure of tables, rows, triggers and functions within the DBMS. So a Data-Dictionary contains all needed technical metadata for a DBMS. The great advantage of such a dictionary is the separation of the metadata from the application data of the DBMS [11].

Thereby, the structure of the dictionary is different in most DBMS. For example, the system tables in Oracle can not be read by users of the system. A user only gets access over defined views of the system tables. There is no general schema of a Data-Dictionary. The advantage of covering metadata and application data over relations or views within a DBMS is the possibility, to access the connected data over a query language like SQL. To avoid inconsistency on the data, all tables in the data-dictionary are set to read-only for the user. Changing data in the dictionary is only implicitly possible by using a system query like *CREATE TABLE* or *ALTER TABLE* that changes the data in the dictionary.



(a) Harness skeleton for metadata (b) Metadata view pattern

Fig. 4. Harness realization

For our approach to use harnesses to connect metadata with application data, we will use similar techniques as used in the design of a Data-Dictionary in a DBMS like DB2 [11]. As described in 3.1, our six metadata categories are represented by components. A harness can be used to connect these metadata components with the application data dimension. Figure 4(a) shows a general harness skeleton that represents this connection. We now map this skeleton on a HERM model that represents it. To find such a model, it is important that every metadata apply only to one category. Therefore, every application data can only have zero till six connections to the metadata components. The number of connections depends on the existing metadata for the application data. Thus, a harness can be mapped to a view that covers all relevant metadata for the description of an application date. The harness will not be used for inserting data directly into the system. Consequently, the harness can be connected to the output views of the components and builds a read-only connection. Metadata modifications are only implicitly possible by using the input views of the corresponding component. Figure 4(b) shows a pattern of a generic realization of a harness into a HERM model that connects application data with the corresponding components from the metadata dimension. Every connection to a metadata category is optional for the relationship type. It is important that such a relationship type is only compiled, if at least one connection to a metadata category exists. Otherwise, such a relationship is dispensable and must not be compiled. So a harness can be mapped on a n-ary relationship type with $n \in \{2, 3, 4, 5, 6, 7\}$ connections. This relationship type represents a metadata view. Such a view has a similar structure, functionality and content as a view in a data-dictionary of a DBMS.

Therefore the quantity of all metadata views in a system builds a metadata data-dictionary. This dictionary covers all metadata that exist for the application data within the system.

Finally, we want to show that our metadata Data-Dictionary represents a metadata repository. In [1], a *metadata repository* is defined as an integrated, virtual area to charge with metadata. Input, access and structure are independent from a special vendor. The repository is used to store metadata and can be used as interface to external metadata. A repository has to meet six requirements:

1. The content of the repository can be connected to every other content within or outside the repository.
2. Input, access and structure are independent from a vendor.
3. The metamodel of the repository is easily extended without effects on the functionality.
4. Every user can search and access the metadata directly without getting irrelevant data.
5. There is a versioning of metadata within the repository.
6. Metadata and the metamodel are protected against unauthorized access.

Our approach of a component-based metadata Data-Dictionary meets all of these six requirements and is a valid metadata repository. The metamodel is based on components. The components are independent from a distributor and can be extended easily. Connections to every data inside and outside our repository can be established by harnesses. Accessing and searching within the metadata Data-Dictionary without irrelevant data is given because every harness in the dictionary can be compiled for a special usage with the needed metadata information. The last two items depend on a good implementation of such a data dictionary. Thus, with a good implementation these requirements are also met.

4 Related Work

The categorization of metadata can be found in different scientific works in literature. In [1], there is a categorization into *specific*, *unique* and *common*. It is based on the assumption that different disjoint user groups can be identified within a system. Specific metadata cover all data that is only created and used by the same user group. If a user group utilizes metadata that is created by another single user group, it is unique metadata. Metadata that is created and used through all user groups in the system is called common metadata. This is an user centric approach that requires the user group identification in a system. Furthermore, there exist metadata that could not be immediately categorized. A categorization that is not user centric can be found in [9]. There, we can also find six different categories of functional metadata; namely *Administration*, *Terminology*, *Context driven*, *Governance*, *Structure* and *Operative*. They should

support the terminology management for master data management. All this categories are also answering special WH-questions.

Beside the categorization of metadata, there exist a lot of standards that take a close look at the field of metadata management. Two of the important standards are the *Dublin Core* and the *Common Warehouse Metamodel* (CWA). The *Dublin Core* is standardised *International Organization for Standardization* (ISO) under ISO 1583 [12]. It covers 15 metadata elements that can be used to describe a resource. The purpose of using the *Dublin Core* is the easy detection of resources within a system. Examples for *Dublin Core* elements are title, type and format of a resource. The *Common Warehouse Metamodel* was standardised by the OMG in 2001. It is the prime standard for the field of data warehousing [2]. The CWA is a component-based approach and covers five layers; namely *Object-Model*, *Foundation*, *Resource*, *Analysis* and *Management*. Every layer consists of packages, and every package consists of components that cover a set of metadata elements for the package. There exist other standards for metadata for different levels. There are meta-meta architectures like the *Meta Object Facility* (MOF). Also there are languages for describing metadata like the *Resource Description Framework* (RDF) or describing the interchange of metadata like the *XML Metadata Interchange* (XMI). Furthermore, there are standards for special areas like the *Learning Object Metadata* (IEE LOM), or the *ISO 19115* for geographic information.

There exist a great set of papers about metadata models for highly specialized problems[2]. But there are only some works about generic metadata models. Most of these generic metadata models are models for special areas. An example of such a generic framework can be found in [13]. The framework is based on XML and describes a metadata format for multimedia data that covers general informations on the one hand, such as the title or the author of a special content, and on the other hand special media data metadata such as the GPS location and the resolution of an image. Additionally, the general information can be extended by standards like the *Dublin Core*. An approach for a generic metadata model that covers any kind of metadata can be found in [14]. This approach is based on a combination of the generic Modelling Principles and the architecture of Data Vaults.

5 Conclusion and Outlook

There is a need for a metadata management system that allows a dedicated management of evolutionary metadata. The approach outlined in this paper is based on components which are used to create a set of six metadata components. These components allow a modularization of the global schema with all advantages as, for example, loose coupling. The components make it possible to construct a metadata dimension that lies orthogonally to the application data and workflow data in the global schema. The metadata dimension can be connected to the other dimensions by using a construct called harness. Harnesses can be created

[2] Due to the limitation of this paper we go without citations of these works.

for the special needs of a user group within a system. All harnesses within a system build the metadata Data-Dictionary. The metadata Data-Dictionary is a generalized version of a database Data-Dictionary. Thus, we get an internal metadata repository that allows an evolution safe dedicated metadata management system that reduces drastically the complexity of the global schema without the disembodiment of metadata into an external system.

We have only outlined a general approach on how a good metadata management has to be modelled. In a subsequent step, we must create conceptual models for the six metadata categories. Such a detailed metadata schema that combines all potential aspects for metadata management is under development. Moreover, we must create a system that implements our approach of a metadata Data-Dictionary. Furthermore, such a system must cover some other problems that can be found, for example, in the field of scientific data management [15] or master data management [3]. All these fields need interfaces for the import of metadata from heterogeneus systems. Also the export of data into external and heterogeneous data massive must be possible in such a system. Topical privacy of data is another part such a system must regard. Privacy covers problems such as the property and disclosure of data inside and outside of such a metadata management system. For all this, our approach should be a first step towards such a metadata management system.

References

1. Tannenbaum, A.: Metadata Solutions: Using Metamodels, Repositories, XML, and Enterprise Portals to Generate Information on Demand. Addison Wesley, Reading (2001)
2. Marquardt, J.: Metadatendesign zur Integration von Online Analytical Processing in das Wissensmanagement. Kovač (2008)
3. Berson, A., Dubov, L., Dubov, L.: Master Data Management and Customer Data Integration for a Global Enterprise. illustriert edn. McGraw Hill Professional (2007)
4. Thalheim, B.: Component Development and Construction for Database Design. Data & Knowledge Engineering 54, 77–95 (2005)
5. Schewe, K.D., Thalheim, B.: Component-driven engineering of database applications. In: Proceedings of the 3rd Asia-Pacific Conference on Conceptual Modelling, APCCM 2006, vol. 53, pp. 105–114 (2006)
6. Thalheim, B.: Entity-Relationship Modeling: Foundations of Database Technology. Springer (2000)
7. Noak, R., Thalheim, B.: Architecturing for Conceptual Modelling in the Large. In: Kiyoki, Y., Tokuda, T., Yoshida, N. (eds.) Proceedings of the 23rd European-Japanese Conference on Information Modelling and Knowledge Bases. Information Modeling and Knowledge Bases XXIII, pp. 29–48. IOS Press (2013)
8. Heinrich, L.J., Stelzer, D.: Informationsmanagement: Grundlagen, Aufgaben, Methoden, vol. 10. Oldenbourg Wissensch. Vlg (2011)
9. Scheuch, R., Gansor, T., Ziller, C.: Master Data Management: Strategie, Organisation, Architektur. Dpunkt. Verlag GmbH (2012)
10. Zachman, J.A. (2008), `http://www.zachmaninternational.com/index.php/the-zachman-framework`

11. Denne, N.: DB2: Theorie und Praxis, vol. 7. DGD-Ed. (2001)
12. Gordon, K., Society, B.C.: Principles of Data Management: Facilitating Information Sharing. British Computer Society (2007)
13. Brut, M., Laborie, S., Manzat, A.M., Sedes, F.: A Generic Metadata Framework for the Indexation and the Management of Distributed Multimedia Contents. In: NTMS, pp. 1–5 (2009)
14. Saratchev, P.: Towards a Generic Metadata Modeling. Yearbook of the Faculty of Computer Science 1(1), 161–174 (2012)
15. Neuroth, H., Strathmann, S., Oßwald, A., Scheffel, R., Klump, J., Ludwig, J.: Langzeitarchivierung von Forschungsdaten: Eine Bestandsaufnahme. Hülsbusch, W (2012)

Intelligent System for Time Series Prediction in Stock Exchange Markets

Nicoleta Liviana Tudor

Department of Information Technology, Mathematics, Physics
Petroleum-Gas University of Ploiesti, Romania
ltudor@upg-ploiesti.ro

Abstract. This article presents an intelligent system using artificial neural techniques for time series prediction in stock exchange markets. For this purpose, is developed a hybrid neural network with supervised learning algorithm able to learn to predict the evolution of stock exchange for a given period of time. The learning model proposed for the intelligent system considers a First Input First Output (FIFO) queue with input values taken from the values obtained by prediction by the neural network at previous time. Analysis of the performance parameters of the neural network uses the method of the coefficient of certainty of neural prediction. Experimental study highlights the effectiveness of the proposed learning model for hybrid neural predictive system properties and its usefulness.

Keywords: intelligent system, hybrid neural network, learning model, uncertain knowledge, performance parameters, time series prediction.

1 Introduction

Learning is a characteristic of intelligent systems, giving them machine learning capabilities. Intelligent systems learn, improving their experience, acquiring or generating new knowledge (see Tudor [14]). Supervised learning is one of the most used techniques by machine learning and can be implemented in artificial neural networks (ANN). An intelligent system can use connectionist type artificial intelligence techniques to learn or to make predictions using supervised learning.

This paper describes an intelligent system designed to predict the evolution of some stock values on Romanian stock market. Bucharest Stock Exchange (BSE) is one of the most active financial instruments of the Romanian economy and aims to build the most efficient capital markets in South East Europe. The stock market aims to facilitate the successful completion of public offerings for public companies, to increase the number of active companies on BSE, to reduce bureaucracy and to implement the best international practices.

In a modern economy, stock market prediction can be performed with financial instruments or by methods using artificial intelligence techniques such as

W. Abramowicz and A. Kokkinaki (Eds.): BIS 2014, LNBIP 176, pp. 122–133, 2014.

neural networks, knowledge based systems or genetic algorithms. Neural solutions provided by the literature to predict the stock market offer a variety of neural solutions such as: feed forward and recurrent neural networks.

Article contribution. This paper provides an innovative approach to the time series prediction problem because it introduces a new model of learning for hybrid neural networks. The hybrid neural network contains layers feed forward layers but, the last layer output is fed back to the network inputs. Therefore, the network becomes a partially recurrent network. The novelty of the proposed learning model for hybrid neural networks is that supervised learning uses a FIFO queue of input values filled in with values known and values the neural network generated by prediction. Experimental study of Romanian stock market prediction results considers market statistics relating to financial instruments such as stocks and evolution of BET (BUCHAREST EXCHANGE TRADING) price index developed by the BSE. A comparative analysis of price index of BSE and American index is also performed.

The paper is organized as follows. Section 2 presents the problem of predicting the time series. Section 3 defines neural network architecture and learning algorithm. In Section 4, the author presents neuronal software system and the learning model the network uses to predict the evolution of the stock exchange. Section 5 deals with the experimental results where are analyzed the tests on the proposed system showing the performance parameters. Then, conclusions are presented in section 6.

2 Related Work

There are a large number of works on time series prediction, such as: autoregressive (AR), moving average (MA), and autoregressive moving average (ARMA) models, which have been successfully used in many applications such as speech analysis, noise cancelation, and stock market analysis. In statistics, an autoregressive (AR) model is a representation of a type of random process. In finance, stock market prediction methods include fundamental or technical analysis and alternative methods (see Azoff [1]). Technical analysis seeks to determine the future price of a stock based on the potential trends of the past price.

Alternative methods for stock market prediction can use hidden Markov models (HMM), feed forward networks with the backpropagation algorithm, or probabilistic neural networks. A HMM is considered the simplest dynamic Bayesian network. Another form of ANN more appropriate for stock prediction is the recurrent neural network or time delay neural network (Cellular neural networks, Elman, Jordan, and Elman-Jordan networks) (see Enke et al. [5]). Latest studies on the prediction of market trends are based on complex algorithms for Web search. This programming technique allows analyzing the behavior of stock exchanges by counting searches on Google site (Tobias et al. [12]).

We extend our work within (Tudor [15]), by introducing a partially recurrent neural network. Neural network performance in the prediction of time series is evaluated in terms of method of certainty coefficient of processed knowledge.

The advantage introduced by creating the partially recurrent network consist of possibility to use a FIFO queue of input values filled in with values known and values the neural network generated by prediction. The input data of the network are considered uncertain knowledge because knowledge generated by prediction during learning algorithm are processed as input data to the next steps.

3 Predicting the Time Series

Time series prediction can be formalized mathematically for a series of values $x_1, x_2, \ldots, x_n$ at n successive points in time. It is assumed that it wants estimating value relating to step $n + 1$. In the event that the value x_i depends on k previous values, $x_{i-k}, \ldots, x_{i-2}, x_{i-1}$, it is considered that it can be designed a neural network trained to determine the relationship between any pair of k consecutive values and the value on the next position, the position $k+1$ (Tudor [13]).

We apply time series prediction theory for predicting the evolution of Romanian stock exchange transactions. Analysis of prediction of stock exchange is made from two perspectives: market statistics relating to financial instruments such as stocks and price index BET(BUCHAREST EXCHANGE TRADING).

BET index is the benchmark index of the capital market. BET is a price index weighted by capitalization of free float of the 10 most liquid companies on the BSE market. From indices of regional markets, BET index of BSE is most strongly correlated with the U.S.A. S&P 500 index.

Financial instruments such as shares. We consider the types of shares from BSE sections, RASDAQ and ATS, the value of shares in currencies RON, EUR, USD, volume and number of transactions and the last update. The intelligent system will process the total number of shares and transactions run for each section of stock market, every working day during a frame of time.

4 Design of the Proposed Neural Network

We design a hybrid network with feed forward layers. The network has the property of being partial recurrent due to a feedback connection of the output of the last layer, which does not influence the typology of connections between internal neurons and the calculation of the outputs of network.

The novelty of the model introduced is that the hybrid network takes input data from a FIFO queue in which the first n values are known, and the other values are obtained by prediction from the network output, where n is given.

4.1 The Hybrid Neural Network Architecture

To highlight the property of prediction BSE stock market values is considered a range $x_1, x_2, \ldots, x_n$ of stock exchange values obtained in successive days and is estimated the value appropriate to step $n + 1$.

A hybrid network model for the prediction of stock market values may have $k(k < n)$ input neurons, a given number of hidden units and a single linear output neuron. Architecture of 2-layer feedforward network has the following characteristics:

- m neurons on the hidden layer and one linear neuron in the last layer
- x_j, $j = 1, \ldots, k$, is the vector with k input signals of network (Fig. 1.)
- w_j, $j = 1, \ldots, m$, is the weight between layers 1 and 2 (hidden layer and neuron in the last layer) (Tudor [15])
- the activation of a neuron is obtained by processing the input signal values which can be the weighted sum of the inputs (Rojas [10])
- f is the transfer function that transforms the activation signal to an output signal
- network training uses a Backpropagation algorithm
- t is the desired output and $a^{(2)}$ is the network output defined as follows:

$$a^{(2)} = \sum_{i=1}^{m} w_i f(a_i^{(1)}) = \sum_{i=1}^{m} w_i f(\sum_{j=1}^{k} w_{ij} x_j)) \; . \tag{1}$$

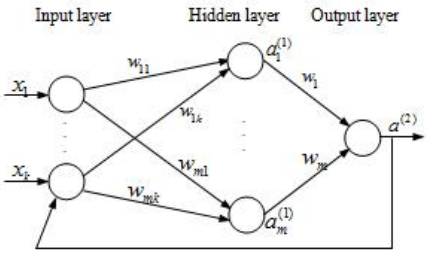


Fig. 1. Hybrid neural network for predicting the stock

For a fixed training set, a minimum hidden layer size (threshold) does exist but it can be detected only by using a suitable learning algorithm and data normalization (Fulginei et al. [6]).

4.2 Supervised Learning Algorithm

Learning method uses supervised learning which implies the existence of a training set consisting of pairs of type (input, desired output) (see Carpenter et al. [2]) and Haykin [8]). Network training uses a Backpropagation algorithm, in which input data use feedback connection of hybrid neural network.

To represent the exchange values using the training set, are considered known values of representative shares for three sections of the BSE stock exchange in

successive days. The training set will contain $n-k$ pairs of the form (known input, desired output) (Tudor [14]):

$$[((x_1, x_2, \ldots, x_k), x_{k+1}), \ldots, ((x_{n-k}, x_{n-k+1}, \ldots, x_{n-1}), x_n)] \; . \tag{2}$$

For each set of known values $x_1, x_2, \ldots, x_n$, neural network approximates x_{n+1} value. In step j $(j < n)$ of the learning algorithm, inputs are represented by the crowd of values $x_j, \ldots, x_{j+k-1}$. The neural network determines by prediction the x_{j+k} value. After n steps, all network entries will come from the outputs of the network obtained by prediction. And so the process is repeated, the network using a recurrent transmission of neural flow.

Hybrid neural network uses a learning algorithm based on Back propagation specific to certain class of dynamic networks (see De Jesus et al. [4]). By training, are updated only direct connections between neurons (see Montavon et al. [9]). During training, the changed values of the weights between neurons store the financial knowledge. Processed knowledge propagates with neuronal flow and future values of exchange transactions are generated by prediction.

Next, we will show the Backpropagation algorithm for the proposed hybrid network using a matrix calculation approach adapted from Rojas (Rojas [10]).

Backpropagation algorithm involves two phases. Forward phase, sets the network parameters and the input signal is propagated through the network, layer by layer. A diagonal matrix with derivatives calculated for hidden layer can be arranged as follows:

$$D_1 = \begin{pmatrix} a_1^{(1)}(1-a_1^{(1)}) & \ldots & 0 \\ \ldots & \ldots & \ldots \\ 0 & \ldots & a_m^{(1)}(1-a_m^{(1)}) \end{pmatrix} \; . \tag{3}$$

Determine derivates for the last layer using network output as follows:

$$D_2 = \left(a^{(2)}(1-a^{(2)}) \right) \; . \tag{4}$$

At the end of this phase, error is calculated for the last layer, as the difference between the inputs produced by the network and the desired outputs (target):

$$E_2 = a^{(2)} - t \; . \tag{5}$$

In Backward phase, the error of the last layer propagates backwards through the network in order to minimize overall training error, ie reaching the stage where the neural network will produce an output signal as close to the desired output.

Training error for neurons in the last layer can be determined as follows:

$$D_2 * E_2 = \left(a^{(2)}(1-a^{(2)}) \right) * (a^{(2)} - t) \; . \tag{6}$$

Training error spreads back, from the last layer to the neurons in hidden layer:

$$D_1 * (w_{i=1,\ldots,m})^T * E_2 \; . \tag{7}$$

5 Case Study: Intelligent System Based on Uncertain Knowledge

The problem of predicting the evolution of Romanian stock market transactions is solved using an intelligent system developed in Matlab, which implements a hybrid neural network with learning algorithm defined above.

Hybrid neural network uses as input data, values stored in a data structure of type queue (FIFO), where the first n values are real data taken from daily transaction reports from Bucharest Stock Exchange. Other values of FIFO queue are obtained by prediction from the output of the network. Neural network output contains predictive generated values of stock exchange in about 98%. Reasons for choosing this type of network is derived from the operation and validation of predictive neural system of evolution of stock values for a given period of time.

5.1 Training Set of Network

To define the network input data, we will consider the following parameters: number of shares (volume) and the number of transactions on the BSE on n consecutive days in the current month when the stock market is active. We assume that the variation of BET index is calculated only for sections RASDAQ and Regulated markets.

For the encoding of the training set of the neural network, we consider the particular case of $n = 9$ succesive known input, 5 training sets ($k = 4$ inputs and 1 output) and the target vector (desired output) has 5 values.

The neural network uses as a data structure a FIFO queue containing the initial $n = 9$ values of the representative shares of sections BSE, RASDAQ ATS, expressed in EUR, known for previous consecutive days. Each output value obtained by network prediction is added to the end of the queue and the first value is removed from the beginning of the queue. In every 9 consecutive values taken from the beginning of the queue, form 5 sets of training sets matrix represented as follows: $[x_1, x_2, x_3, x_4, x_5; x_2, x_3, x_4, x_5, x_6; \ldots; x_5, x_6, x_7, x_8, x_9]$. At each step, the network determines x_{10}.

5.2 Certainty Coefficient Method

Neural network performance in prediction of exchange transactions evolution is influenced by the accuracy and veracity of the network output. Analysis of prediction efficiency will consider the sources of uncertainty in the neural system, such as language translation of financial knowledge in neuronal size of the system, knowledge missing in the chain of reasoning or factors that depend on the activity of the stock market (fluctuations between maximum and minimum values of the shares), etc.

Common methods of uncertain knowledge representation in intelligent systems include fuzzy model, certainty rating method, and Bayesian neural network

method (see Tudor [14]). To study the proposed neural network performance, we use certainty coefficient method, which measures the degree of veridicity of knowledge.

Evaluation of the propagation of uncertain knowledge through neuronal flow can be achieved using confidence rating method. Coefficient of certainty will be made on logical and objectives principles, with intuitive and empirical character. Several relevant factors such as the average difference between the known values of the parameter number of transactions and those approximated by prediction noted $Mdif$ and maximum and minimum values of the parameter noted Max and Min will be considered (Table 1).

Table 1. Performance parameters of hybrid neuronal network

Number neurons/ hidden layer	Activation function	Mean network error	Average differences/ parameters	Certitude of network output data (percent)
50	Tansig	0.48	1513	1.5
70	Tansig	0.43	639.23	44.70
150	Tansig	0.56	1932	1
50	Logsig	0.60	399.02	65.48
70	Logsig	0.67	594.43	48.57
150	Logsig	0.65	1131	2.16

To determine the coefficient of certainty of the knowledge obtained by prediction we introduce a heuristic function f:

$$f(x) = 100 - Mdif(x) * 100/(Max(x) - Min(x)) \; . \tag{8}$$

where x is the vector of transactions during the period studied (known values and predicted values). Exceeding a limit of confidence rating (ie [0, 100]) means that this knowledge is false. For the experimental study, will be selected the network parameters that generate maximum value of the heuristic function f.

5.3 Intelligent System Performance

For the analysis of hybrid neural network performance, we can use a cost function. During learning, neural network weights and thresholds are adjusted to minimize the cost function (see Montavon et al. [9]). If the cost function is defined using a vector, the network optimum is a local minimum of the function. In other words, the learning algorithm is executed until the cost function (or network error) reaches a predefined minimum. Initialization of weights and thresholds of a network with small values randomly generated may influence the convergence of the cost function to a local minimum (see Dumitrescu et al. [3]).

Cost function enables determining the average error of network performance and depends on parameters such as: number of neurons in the hidden layer, activation function used in the hidden layer, learning rate, the learning algorithm with/without momentum. The functioning of the neural system is tested for learning rate values between 0.05 and 0.4, the number of neurons in the hidden layer varies between 50 and 150. Because sigmoid functions have the non-linearity and differentiability advantage, the tansig and logsig activation functions are used for the neurons of the hidden layer (see Gheorghita et al. [7]).

Mean absolute error for the overall evaluation of neural network is calculated as the arithmetic average of the mean square error obtained at each subinterval of the division which apply neural networks. We obtain a good average error neural network for a large number of neurons, a training function sigmoid tangent type for the hidden layer and for a Backpropagation algorithm without momentum.

For a better prediction of stock market, we can use heuristic methods to optimize learning algorithm. If training rate varies at each step, we obtain a generalized variant of the error back propagation algorithm (see Won [16]). Sejnowski and Rosenberg have shown experimentally that learning rate and hidden neurons number may increase with network performance (Sejnowski [11]). Also initialization of network weights and thresholds with low values randomly generated can optimize the learning algorithm (see Dumitrescu et al. [3]).

Intelligent system performance can be demonstrated by evaluating the difference between the values approximated by neural network prediction and the real values obtained in the BSE in prediction period, that means the last 6 days of the 15 days studied. To analyze the efficiency of stock market evolution prediction, determine the value of an accuracy function noted Acc defined for each section x of the stock market in the prediction period as follows:

$$Acc(x) = \frac{1}{2} * \sum_{k=1}^{2} |(Max_k - Min_k)/Dif_k| \ . \tag{9}$$

where Max_k (respectively Min_k) is the maximum (or minimum) value of number of shares/transactions in prediction frame time; calculated daily Dif_k is the difference between actual and predicted number of shares/transactions.

6 Experimental Results

Testing and validating neural system is distinguished by a comparative analysis of the following types of parameters: (a) the number of shares traded, (b) the number of transactions and (c) BET Bucharest Stock Exchange index for three sections stock market, each day of the study.

6.1 Study of Performance Parameters

Parameter: number of shares traded. The study of evolution of the number of shares traded on BSE in the current month uses the known values of the number

Table 2. Prediction of BSE shares evolution for the three sections

Day	Regulated market	RASDAQ	ATS	Day	Regulated market	RASDAQ	ATS
10	10000000	1000000	10000	20	61409000	7213400	2168
11	20592000	14657	3466	23	14548000	515000	56162
12	41539000	1465700	1415	24	15466000	434900	56038
13	22686000	230700	430	25	22256000	455000	55416
16	26383000	57000	350	26	17854000	416300	554859
17	52898000	182500	804	27	24594000	241400	54648
18	13387000	315600	8600	30	21040000	127300	54485
19	20451000	243400	186				

of shares for the three sections of stock in the first $n = 8$ days. Neural network approximates by predictions the values of BSE shares since the ninth day of the 15 days series studied (Table 2).

Parameter: number of transactions performed. The analysis of the evolution of the current month transactions BSE considers the number of transactions known for the three sections of stock in the early $n = 8$ days. Prediction of BVB transactions values starts from the ninth day of the 15 days series (Fig. 2).

Parameter: BSE BET index. The study of the evolution of the BSE BET for current month uses known values of the index for $n = 8$ days and prediction values obtained by neural networks for the remaining days of the month. Consider that during the period studied, the BET-EUR is approximated by neural prediction for two stock sections: the regulated market and RASDAQ. The intelligent system developed in Matlab provides sets of BET values for the two types of shares (Fig 3).

Comparative analysis of the evolution of BET for two sections of the stock is evidenced in Fig 4.

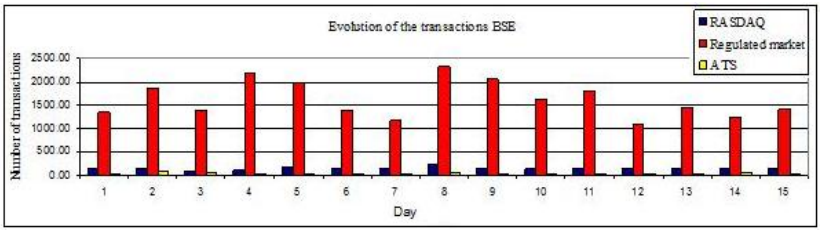


Fig. 2. Evolution of the number of transactions for the three BSE sections

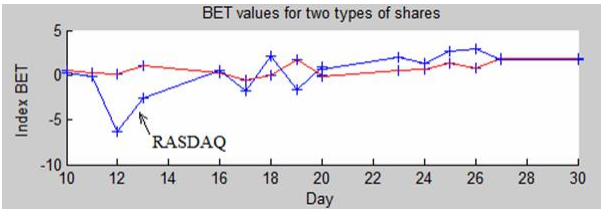


Fig. 3. BET-EUR prediction for two sections: the regulated market and RASDAQ

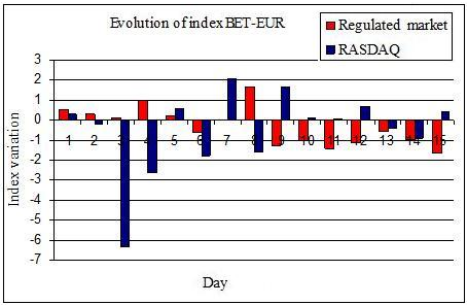


Fig. 4. Analysis of the evolution of BET-EUR for two sections

6.2 Comparison with American Index Standard and Poor's 500

That means a comparison between the average change in percent of the BSE BET in EUR currency for the two exchanges calculated with Romanian Nasdaq index variation and U.S. Standard&Poor's 500 (S&P 500) index for the period studied.

To evaluate the sensitivity of regional markets in the U.S. stock market evolution we may determine a correlation coefficient of BET and Nasdaq with American index S&P 500.

But for the period studied, we analyze the correlation coefficient of BET-EUR, Nasdaq and S&P 500 indices, evaluating the total variation of each index. Results of analysis show that for an increase of 7.32% for the BET-EUR index over a period of 9 days, you get a drop in U.S. index S&P 500 in average of 1.04% and the Nasdaq drops with 0.06 (Table 3). This results in a better correlation between U.S. index S&P 500 and Nasdaq index than with the BET-EUR.

Table 3. Correlation coefficient of stock indices

Mean index BET-EUR variation (%)	S&P 500 index variation (%)	Nasdaq index variation (%)
7.32	−1.04	−0.06

6.3 Intelligent System Performance

System performance is demonstrated by evaluating the difference between the values approximated by neural network prediction and the actual values obtained in the BSE in prediction period, that means the last 6 days of the 15 days studied. To analyze the efficiency of stock market evolution prediction, we must determine the value of an accuracy function noted Acc defined for each section of the stock market in the prediction period.

The comparative analysis of accuracy function of stock prediction is illustrated graphically in Figure 5. Accuracy function of stock prediction provides minimum values for stock sections of Regulated Market and RASDAQ in the last 5 days before prediction, which means a very large nearby true values obtained for BSE transactions. For ATS stock section is obtained a weaker prediction in the last 5 days against the first 2 days of the period of prediction. To increase the system performance it can be considered other performance parameters that can influence the evolutionary prediction of transactions such as financial instruments (Debt Securities, Market Capitalization), etc.

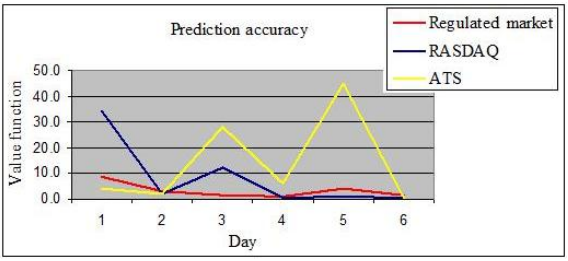


Fig. 5. Analysis of the evolution of BET-EUR for two sections

7 Conclusions

The article presents a model of intelligent system using neural techniques for time series prediction. For system design is created a hybrid neural network that generates by prediction the stock market values for the future close study period.

The basic idea is that we start from a fixed set of known values for BSE exchange shares for a given period and neural network predicts the following values. The hybrid supervised learning model borrows from feed forward networks the Backpropagation algorithm idea, holding the benefit of properties of feedback connections. The proposed learning algorithm performance is evaluated using the coefficient of certainty method to determine the accuracy and reliability of the values generated by hybrid neural network prediction.

The conclusion of the study is that the intelligent system based on a hybrid neural network architecture performs an effective prediction of stock market evolution. Values obtained by prediction are close to the real ones, and BET

may have, in some periods, a lower correlation rate with U.S. S&P 500 index and Nasdaq. In future studies, we can investigate the influence of uncertainty upon learning model effectiveness of hybrid neural network using adaptive techniques.

This study may be further improved by taking into consideration a similar idea for BSE shares prediction, using Markov models (prediction by partial matching) or other statistical methods.

References

1. Azoff, E.M.: Time Series Forecasting of Financial Markets. Neural Network, John Wiley and Sons Ltd. (1994)
2. Carpenter, G.A., Grossberg, S., Arbib, A.M.: The Handbook of Brain Theory and Neural Networks, 2nd edn. MIT Press, Cambridge (2003)
3. Dumitrescu, D., Costin, H.: Neural networks. Theory and applications. Teora Publishing House (1996) (in Romanian)
4. De Jesus, O., Hagan, M.: Backpropagation Algorithms for a Broad Class of Dynamic Networks. IEEE Transactions on Neural Networks 18(1) (2007)
5. Enke, D., Thawornwong, S.: Forecasting Stock Returns with Artificial Neural Networks. In: Zhang, G.P. (ed.). IRM Press (2004)
6. Fulginei, F.R., Laudani, A., Salvini, A., Parodi, M.: Automatic and Parallel Optimized Learning for Neural Networks performing MIMO Applications. Advances in Electrical and Computer Engineering 13(1), 3–12 (2013)
7. Gheorghita, S., Munteanu, R., Graur, A.: An Effect of Noise in Printed Character Recognition System Using Neural Network. Advances in Electrical and Computer Engineering 13(1), 65–68 (2013)
8. Haykin, S.: Neural Networks: A Comprehensive Foundation, 2nd edn. Prentice-Hall, Englewood Cliffs (1999)
9. Montavon, G., Orr, G.B., Müller, K.-R. (eds.): Neural Networks: Tricks of the Trade, 2nd edn. LNCS, vol. 7700. Springer, Heidelberg (2012)
10. Rojas, R.: Neural Networks A Systematic Introduction. Springer, Berlin (1996)
11. Sejnowski, T.J., Rosenberg, C.R.: Parallel Networks that Learn to Pronounce English Text. Complex Systems 1 (1987)
12. Tobias, P., Moat, H.S.: Stanley, H. E.: Quantifying Trading Behavior in Financial Markets Using Google Trends. Scientific Reports 3: 1684 (2013)
13. Tudor, N.L.: Neural networks. Matlab applications. MATRIX ROM Publishing House Bucharest (2012) (in Romanian)
14. Tudor, N.L.: Logic programming and expert systems. Visual Prolog and Exsys applications. MATRIX ROM Publishing House Bucharest (2012) (in Romanian)
15. Tudor, L.: Intelligent system based on supervised learning for predicting the evolution of stock exchange transactions. In: 22nd IBIMA International Business Information Management Conference, Italy, pp. 1128–1134 (2013)
16. Won, Y., Gader, P.: Morphological Shared-Weight Neural Network for Pattern Classification and Automatic Target Detection. Recognition, Electronics and Telecommunications Research Institute, Daejon (1995)

Natural Language Processing for Biomedical Tools Discovery: A Feasibility Study and Preliminary Results

Pepi Sfakianaki[1], Lefteris Koumakis[2], Stelios Sfakianakis[2], and Manolis Tsiknakis[1,2]

[1] Department of Informatics Engineering,
Technological Educational Institute, Heraklion, Crete, Greece
[2] Computational Medicine Laboratory, Institute of Computer Science,
Foundation for Research & Technology – Hellas (FORTH), Heraklion, Crete, Greece
pepsfak@gmail.com, {koumakis,ssfak,tsiknaki}@ics.forth.gr

Abstract. Discovery of the appropriate computational components, needed to answer a clinical hypothesis, has been a major issue for physicians. Users without experience do not have the means, the time or the knowledge to search the vast amount of information regarding the candidate computational components (services or tools) which can aid to achieve their purpose. In order to address this need we introduce a dynamic service discovery environment where physicians can represent queries in natural language and dynamically retrieve the suitable candidate computational components, with the aid of information extraction algorithms guided by specific domain ontologies.

Keywords: linguistic cognition, natural language processing, information extraction, medical entity recognition, relation annotation.

1 Introduction

Nowadays a wealth of publicly available biomedical resources, such as data, tools, services, models and computational workflows, exists. The growth of biomedical data gets the researchers in front of impediments in identifying the resources they need, among the huge range of the accessible resources. Only in the genetics and biology domain, the Biocatalogue[1] online repository lists more than 2350 tools that have been implemented as web services. Unfortunately locating the tools needed for answering a clinical question in a clinical setting or in medical daily practice is often extremely difficult especially for non IT savvy clinical users. The reason is twofold: On the one hand, most of the available tools are annotated with technical details that, although relevant (for example medical ontology terms), are not easily interpreted and "absorbed" by the end users. On the other hand, many end users, especially medical doctors and clinicians, prefer to formulate their queries using natural language since it is quicker and more intuitive and natural.

Of course a natural language interface presents a lot of technical challenges. Computers are good at processing structured data but much less effective in handling

[1] https://www.biocatalogue.org (accessed 03/01/2014).

W. Abramowicz and A. Kokkinaki (Eds.): BIS 2014, LNBIP 176, pp. 134–145, 2014.

natural language which is inherently unstructured. The field of natural language processing [1] (NLP) has been aiming to narrow that gap.

Software providers are approaching ontologies and terminologies to annotate their systems and publish them in specialized repositories. The aim of a repository is to make software more visible, better documented and easier to locate and use. The software components can be described and searched in multiple ways based upon their technical types, categories, description, user tags or data inputs and outputs. Structuring the information of a software component (metadata), mainly using descriptions for the functionality and the parameters, can give us better search and retrieval results. Metadata is data about data, namely descriptions of information; the "where, who, what, how, when and why" about the data. The importance of metadata derives from the facts that it facilitates discovery of repository content, helps organization of the content in a repository, facilitates external systems to be fed with the repository content and supports classification and maintenance of information. A common approach to structure metadata is to apply special ontological concepts for the characterization and annotation of the information. Ontologies provide a formal representation of knowledge as a set of concepts and the relationships between the concepts within a domain.

The mission of service discovery is to seek an appropriate service on the basis of the service descriptions in dedicated repositories. The main burden of an efficient service discovery mechanism is that the standard language used for encoding service descriptions does not have the capacity to specify the capabilities of a service or the interpretation of them from the end user, leading to the problem of ambiguity in the service discovery process. In the literature we can find many methodologies dealing with this problem but only in a well-defined closed domain which is the world of the web services. Priyadharshini et al [2] provided a systematic review on the existing approaches for the discovery of semantic web services and identified six main categories of such tools, while only one out of the six can handle natural language queries, naming context aware discovery, which is described as useful for request and result optimization and personalized service discovery. We have to acknowledge that even for web-services, service discovery is a complicated task especially when the end user is not an IT expert. Posing queries in natural language and identifying possible tools or operations able to provide the requested functionality is a task which requires advanced NLP techniques combined with efficient service discovery mechanisms.

The objective of this paper is to explore semantics (metadata descriptions) used for biomedical resources, such as tools, applications, services, web services and models, through natural language processing capabilities and enable end users - non IT experts - to write descriptions in natural language which can be transformed automatically into patterns enriched with ontology terms for efficient dynamic service discovery. We compare three different methods for natural language processing in medical text to retrieve relevant tools: (i) using free text search and patterns, (ii) using tags from annotated terms of domain ontologies and patterns and (iii) using semantic types of annotated terms (generalization) of the domain ontologies and patterns.

An analysis of the system's implementation methodology is presented in Section 2. Section 3 describes the experiments on clinical texts and comparisons that we were able to realize and in Section 4 we make our conclusions and discuss future work.

2 Methodology

This study focuses on the interpretation of the clinician's requests (posted in natural language) to targeted queries in a service repository for efficient service discovery. For that purpose we use the service repository of the P-Medicine[2] European project as backend. The tools stored in the repository are semantically annotated with multiple ontologies from two different domains providing metadata about descriptions and input/output parameters of the tools. These two domains are (i) the software domain using the EDAM ontology [3] and (ii) the biomedical domain using the Unified Medical Language System (UMLS) Metathesaurus [4], Using this kind of contextual information we aim to facilitate more intelligent search results that address what a user is actually looking for, rather than simply returning any tool that matches the given keywords.

2.1 Architecture

The setup of the system is based on three main components: (i) the tools repository, (ii) the semantic annotator ("Tagger") and the intelligent engine ("Interpreter") which interacts with the end user. As shown in figure 1, the end user interacts with the system via a web based user interface.

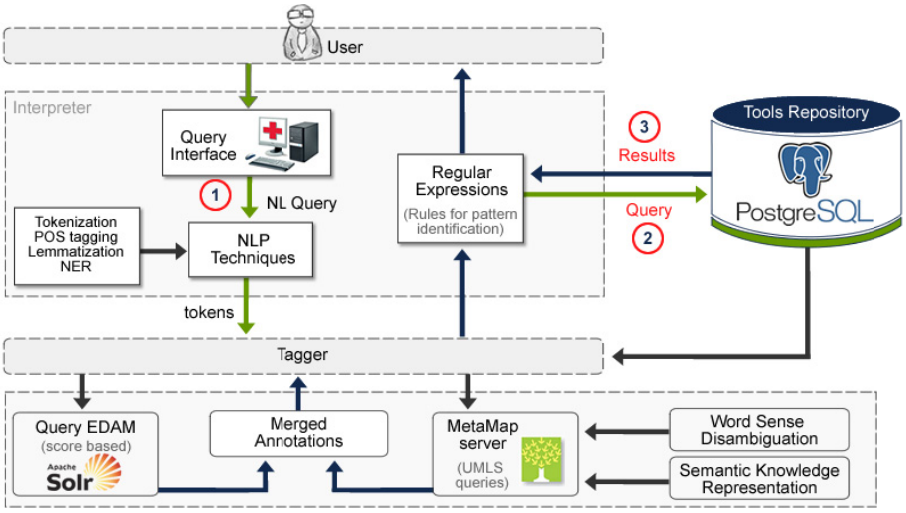


Fig. 1. System's architecture

[2] http://p-medicine.eu/ (FP72007-2013 grant agreement N° 270089).

The user writes the clinical question he/she wants to explore using special tools, e.g. "*Jonh is 5 years old and was diagnosed with wilm's tumor. Given his gene expression I would like to know if carboplatin can make the tumor smaller before surgery and which are the interactions of carboplatin*". The interpreter parses the natural language text using NLP pre-processing algorithms for tokenization, lemmatization and part of speech. Then the system communicates with the Tagger server and extracts ontology terms, concepts and semantic types. Using the ontology terms and its semantic types the system identifies predefined patterns and can post focused queries to the repository of tools/services for an efficient service discovery. Finally the system proposes tools/services to the end user.

2.2 Tagger

In the literature we can find methodologies from the early 90s that describe annotations or entity-relationship models for clinical text [5]. Nowadays a wealth of applications capable to annotate clinical text into ontologies exists. We cannot say the same for the software descriptions. Since our system aims to identify bioinformatics software to solve a clinical hypothesis, we needed and implemented a special tagger for the EDAM ontology. EDAM (EMBRACE Data and Methods) is an ontology of bioinformatics operations, types of data and identifiers, application domains and data formats. EDAM applies to organizing and finding suitable tools and data and to automating their integration into complex applications or workflows. It includes over 2200 defined concepts and has successfully been used for various annotations and implementations. The EDAM tagger implemented using the Solr Apache [6] full text search server. For each term in EDAM ontology we created a JSON-formatted[3] file with specific fields that was subsequently imported to Solr. Solr give us the ability to create custom similarity matching algorithms using weights for each field specified in JSON schema. Our weight formula is predisposed to the *id* and the *name* of the term meaning that if the text which we search matches to the id or the name of a term that term gets a better score than if the text matches to the definition or the comment fields.

The formula of our custom made EDAM weight is:

$$id*10 + name*10 + synonym*6 + subset*3 + is\_a*3 + def*2 + comment*1$$

For the annotation of biomedical concepts we use the MetaMap [7] server which is based on the UMLS. The UMLS Metathesaurus is a unit of files and software that unifies a lot of health and biomedical vocabularies and standards. It combines a variety of source vocabularies, ontologies and terminologies by integrating them into its three knowledge sources: Metathesaurus, Semantic Network and SPECIALIST lexicon. This integration results in a very large medical knowledge base, covering numerous themes in the medical domain. The 2013AA release of UMLS has been used in our system which includes more than 2.8 million concepts and 11.2 million terms from over 160 source terminologies. Our biomedical tagger recognizes and

[3] JavaScript Object Notation, `http://json.org`

annotates the occurrences of medical terms in free texts based on a subset of terminological resources included in UMLS Metathesaurus namely Mesh, Snomed-CT, Loinc, ICD10-CM, RXnorm, MedlinePlus, NCI, OMIM and GO.

The tagger reports also the semantic types, a high level category, of the matched term which provides a consistent categorization of all concepts present in UMLS Metathesaurus or in EDAM ontology. UMLS provides many semantic categories, from which we selected 58 of them and categorized in 23 semantic types naming: Disease, Drug, Symptom, Medical Procedure, Organism Function, Biomedical, Cell, Clinical Attribute, Device, Diagnosis, Gene, Molecular Sequence, Injury or Poisoning, Laboratory, Tissue, Temporal Concept, Age, Body Part, Food, Patient, Virus, Vitamin, Finding. We added four more categories of the Edam ontology: Edam Data/Format, Edam Operation, Edam Topic and the Edam Identifier.

When the end user posts a question, the tagger applies NLP pre-processing algorithms and invokes the 2 domain specific taggers. The EDAM tagger and the MetaMap tagger also support the pre-processing NLP steps. The results of the two taggers are merged. Even though the software and the biomedical domains are independent, some terms can co-exist in EDAM and in one or more UMLS ontologies. An indicative example is the tagging of the text "gene expression" which exists in EDAM ontology (data: 2579) and in Mesh ontology (C0017262). In such cases the tagger merges all the proposed concepts and sends them back to the interpreter as a list of proposed concepts.

2.3 Tools Repository – Import of Data

The tools repository is based on the PostgreSQL database with full text search capabilities. The repository supports three different levels of metadata for the tools, the operations and the parameters (inputs/output). The first level of the supported metadata is the description as plain text. The second level of the metadata consists of the user defined tags as plain text and at the third level are the semantic terms which are concepts from one of the supported ontologies. Currently the repository stores information for 490 tools extracted from different domain specific repositories and web search, as follows:

- 195 tools from the Seqanswers[4], a sequence analysis repository,
- 35 tools from the Embrace[5], a biomedical tools repository,
- 133 tools from Bioconductor[6],
- 75 tools from myExperiment[7] (tools and workflows),
- 50 medical tools and 50 biology related tools by searching the web.

The semantic terms have been generated automatically using scripts which take the textual description of the tool and using the tagger extract ontology terms and semantic categories.

[4] http://seqanswers.com/wiki
[5] http://www.embraceregistry.net/
[6] http://www.bioconductor.org/
[7] http://www.myexperiment.org/

2.4 Interpreter

The interpreter is the bridge between the clinical question and the query in the tools repository. Interpreter uses NLP techniques and the Tagger server to form targeted queries in the tools repository. In our study we used annotated corpora from EDAM ontology and the UMLS Metathesaurus. The framework encompasses a keyword-based search process of Bio-medical tools that are labeled using the same ontologies.

2.4.1 Natural Language Processing

NLP techniques used in the interpreter initiate with the sentence splitting. Then the tokens are recognized among the sub-sentences which are usually words or other atomic parse elements. Once the tokens are extracted, we can identify parts-of-speech and also lemmatize the tokens to the root of the words. Parsing or syntactic analysis, which is the recognition of noun and verb phrases of the text, is then applied.

The analysis of the clinical text provides tokens that are syntactically annotated and therefore in the next step these can be recognized as entities and match with the rules that are connected to the output's objective.

2.4.2 Regular Expressions

The final step of our NLP analysis encompasses regular expressions [8]. A regular expression is a sequence of entities that forms a search pattern. Our system identifies patterns over clinical questions, employed by extraction rules for matching expressions. We applied regular expressions over the clinical question in order to extract rules and relationships between clinical entities. Primarily, we made replacements on the tokens that had annotations, with the semantic types of the tagged terms. Patterns to identify clinical name entities were not needed, because we already had the semantic type of each token. The system also tries to identify and categorizes parts of the input text as input/available data and parts that compose the clinical hypothesis (clinical question to be answered). This is also addressed by regular expressions and pattern identification using special keywords or patterns and with the aid of the MetaMap WSD server.

Table 1. The combined rules

Input & Output Rules	
PATIENT TOOK DRUG FOR DISEASE IN BODY_PART	PATIENT TOOK DRUG FOR DISEASE
DRUG FOR SYMPTOM	DRUG & DRUG FOR DISEASE
DRUG FOR DISEASE	DRUG FOR DISEASE IN BODY_PART
DRUG FOR EDAM_DATA IN BODY_PART	PATIENT HAS DISEASE IN BODY_PART
DRUG FOR EDAM_DATA	PATIENT TOOK DRUG FOR BODY_PART
EDAM_DATA IN BODY_PART	PATIENT HAS DISEASE
SYMPTOM of MEDICAL_PROCEDURE	DISEASE IN BODY_PART
FINDING IN|WITH MEDICAL_PROCEDURE	PATIENT'S FINDING
FINDING with ORGANISM FUNCTION	PATIENT TOOK VITAMIN
PATIENT TOOK DRUG FOR DISEASE IN BODY_PART	PATIENT HAS SYMPTOM
EDAM_DATA or FORMAT and EDAM_OPERATION and EDAM_DATA	PATIENT ATE FOOD
EDAM_DATA or FORMAT and EDAM_OPERATION	PATIENT HAS BEEN ON MEDICAL PROCEDURE
23 UMLS	PATIENT HAS ORGANISM FUNCTION
4 EDAM	

We created a specific rule for every semantic category supported by our *Tagger*. 23 rules for the UMLS semantic types were created and 4 categories for the Edam types. Having these 27 primer rules we created 25 new rules using combinations as shown in Table 1. The same rules were used to get the input and the output parts of the clinical question.

3 Experiments

Facilitating satisfaction of information retrieval activities such as the ones listed in the previous section means that the supported infrastructure should be able to empower users by enabling them to seamlessly post complex queries and to receive accurate results on an ad-hoc basis.

It is hard to evaluate such a system because no standard or benchmark exists for biomedical tools discovery. For our preliminary experiments we created 2 simple clinical questions based on descriptions of clinical trials from the ClinicalTrials.gov registry[8]. For each sentence (clinical question) we identified manually (screening of the tools descriptions in the repository) the tools which could solve or partially solve the specific clinical question. Of course more than one tools can solve the same question. For our experiment, each clinical question passes from the interpreter which identifies the tagged terms of the text with the semantic types based on the ontologies, using the Tagger. The system spits and categorizes the text into the input part (given information/data) and the output part (clinical hypothesis – question) using regular expressions. At the final step interpreter posts specific questions, in most of the cases targeted to the input description of the tools repository. To assess the efficiency of the system we applied 4 different queries to the tools repository starting from full text search:

1. ***Full text*:** we set as query the initial clinical question.
2. ***Tags***: we set as query combinations of the recognized ontology terms (tags) which have been identified in the input or the output part of the sentence.
3. ***SemTypes*:** we set as query the recognized semantic types which have been identified in the input or the output part of the sentence.
4. ***EDAM Data tags for tool inputs*:** we set as query only the identified terms of the categories EDAM data and EDAM format which have been identified in the input part of the sentence and the query is focus only in the description of the inputs of the tools.

The following sub-sections describe the two clinical questions and the results of the system.

3.1 First Clinical Question

The first clinical question for our experiment is: *"John is 5 years old and was diagnosed with wilm tumor. Given his gene expression I want to categorize the*

[8] `http://clinicaltrials.gov/`

patient to the stage of wilm tumor." For this question we identified manually 50 tools (6 tumor specific and 44 gene expression analysis) out of the 490 tools in the tools repository which could solve or partially solve the specific clinical question.

The sentence initially passes from the Tagger which proposes ontology annotations. The annotations for the first clinical question are shown in table 2.

Table 2. The tagged terms' names, the URI of each concept and its category for the first clinical question

Original Text	Ontology terms	Concept URI	Category
wanted	Wanted	http://linkedlifedata.com/resource/umls/id/C1444647	Finding
wilms tumor	WILMS TUMOR	http://linkedlifedata.com/resource/umls/id/C2697327	Gene
year	years	http://linkedlifedata.com/resource/umls/id/C0439234	Temporal Concept
diagnosed	Diagnosed	http://linkedlifedata.com/resource/umls/id/C0011900	Finding
wilm tumor	Wilm's Tumor	http://linkedlifedata.com/resource/umls/id/C0027708	Disease
tumor	Tumor annotation	http://edamontology.org/data_2217	Edam Data
given	Given	http://linkedlifedata.com/resource/umls/id/C1442162	null
gene expression	Gene expression	http://linkedlifedata.com/resource/umls/id/C0017262	Molecular Sequence
patient	*^patient	http://linkedlifedata.com/resource/umls/id/C0030705	Patient
stage	Stage	http://linkedlifedata.com/resource/umls/id/C1300072	Clinical Attribute

Then, the system spits and categorizes the text into the input part (given information/data), the output part (clinical hypothesis – question) or the general part (none of the previous two) using regular expressions. The input and the output parts for our clinical question are shown in Table 3.

Table 3. The input and output parts of the first clinical question

INPUT	OUTPUT
DISEASE : wilm tumor	DOCTOR'S OBJECTIVE: Given his MOLECULAR_SEQUENCE I want to categorize the PATIENT to the stage of DISEASE.
TEMPORAL_CONCEPT : year	OUTPUT: PATIENT HAS DISEASE -> DISEASE : wilm tumor
FINDING : diagnosed	MOLECULAR_SEQUENCE : gene expression

Having categorized the initial text into parts we apply the four different queries to the tools repository (full text, tags, semTypes and EDAM data tags for tools input). Table 4 shows the input for the query, the number of the retrieved tools, the number of the retrieved tools which are actually related to the clinical question (based on the manual selection) and the method of search that we used.

Table 4. The results of the queries made in the tools repository for the first clinical question

INPUTS	Number of TOOLS search selected	Number of related TOOLS manually selected	Method
Jonh is 5 years old and was diagnosed with wilm tumor. Given his gene expression I want to categorise the patient to the stage of wilm tumor.	167	50	Full text
Wilm tumor & year & diagnosed	0	0	Tags
Patient & Wilm tumor	0	0	
Gene expression	19	44	
Finding & Edam data	6	0	SemTypes
Finding	2	8	
Disease & Temporal concept & Finding	0	0	
Molecular sequence	24	12	
http://edamontology.org/data_2217	1	1	Edam Data Tags for tools input

3.2 Second Clinical Question

The second clinical question is: *"John is 5 years old and was diagnosed with wilm's tumor. Given his miRNA I would like to know if carboplatin can make the tumor smaller before surgery and which are the interactions of carboplatin."*

For this question we identified manually 55 tools (6 tumor specific, 24 drug related and 23 miRNA related – tools overlap in these three categories) out of the 490 tools in the tools repository which could solve or partially solve the specific clinical question.

The sentence initially passes from the Tagger which proposes ontology annotations. The annotations for the first clinical question are shown in table 5.

Table 5. The tagged terms' names, the URI and its category for the second clinical question

Original Text	Ontology terms	Concept URI	Category
known	known	http://linkedlifedata.com/resource/umls/id/C0205309	null
year	years	http://linkedlifedata.com/resource/umls/id/C0439234	Temporal Concept
diagnosed	Diagnosed	http://linkedlifedata.com/resource/umls/id/C0011900	Finding
wilm tumor	Wilm's Tumor	http://linkedlifedata.com/resource/umls/id/C0027708	Disease
tumor	Tumor annotation	http://edamontology.org/data_2217	Edam Data
tumor	Tumor	http://linkedlifedata.com/resource/umls/id/C0027651	Disease
given	Given	http://linkedlifedata.com/resource/umls/id/C1442162	null
gene expression	Gene expression	http://linkedlifedata.com/resource/umls/id/C0017262	Molecular Sequence
carboplatin	Carboplatin	http://linkedlifedata.com/resource/umls/id/C0079083	Drug
smaller	Smaller	http://linkedlifedata.com/resource/umls/id/C0547044	null
surgery	Surgery	http://linkedlifedata.com/resource/umls/id/C0038894	Biomedical
mirna	miRNA	http://linkedlifedata.com/resource/umls/id/C1101610	Molecular Sequence
miRNA	Gene ID (miRBase)	http://edamontology.org/data_2642	Edam Data

Then, the system spits and categorizes the text into the input part (given information/data), the output part (clinical hypothesis – question) or the general part (none of the previous two) using regular expressions. The input and the output parts for our clinical question are shown in Table 6 and the queries to the tools repository with the results of the tools can be found in Table 7.

Table 6. The input and output results for the second clinical question

INPUT	OUTPUT
EDAM_DATA : tumor	DOCTOR'S OBJECTIVE: Given his EDAM_DATA I would like to know if DRUG can make the EDAM_DATA smaller before DIAGNOSIS and which are the interactions of DRUG.
TEMPORAL_CONCEPT : year	DRUG : carboplatin
FINDING : diagnosed	DIAGNOSIS : surgery
	EDAM_DATA : tumor
	EDAM_DATA : miRNA

Table 7. The results of the queries made in the tools repository for the second clinical question

INPUTS	Number of TOOLS search selected	Number of related TOOLS manually selected	Method
Jonh is 5 years old and was diagnosed with wilm's tumor. Given his miRNA I would like to know if carboplatin can make the tumor smaller before surgery and which are the interactions of carboplatin.	87	55	Full text
tumor & year & diagnosed	0	0	Tags
drug	5	24	
surgery	0	0	
tumor	0	6	
miRNA	14	23	
Drug & surgery & tumor & miRNA	19	55	
Temporal concept & Finding & Edam data	6	8	SemTypes
Drug	5	24	
Diagnosis	1	2	
Edam data	4	4	
Drug & Diagnosis & Edam data	10	26	
http://edamontology.org/data_2642	7	7	Edam Data Tags for tools input
http://edamontology.org/data_2217	1	1	

4 Discussion – Conclusions

Service discovery is a challenging task not only due to the plethora of the available services and the diversity of their implementation but also for well-defined subsets of services like the domain of web services [2]. As pointed out by Henninger [9] there is a conceptual gap between the problem and the solution, since usually software is described in terms of functionality (how), and queries are formulated in terms of the problem (what).

The use of natural language interfaces for querying structured information goes back to mid-1980s with early attempts focusing on accessing information stored in databases [10] [11]. In recent years there are a few attempts to provide a natural language interface for the semantic web [12], [13] and to evaluate such interfaces [14]

with encouraging results. Nevertheless, to our knowledge, systems that take advantage of NLP techniques for identification of services exist only for semantic web services like Sangers et al [15] which use a keyword-based discovery process for web services described with a semantic language or GODO [16] which uses a goal-driven approach. Their search mechanisms incorporate NLP techniques to establish a match between a user search query and the Web Service Definition Language of a web service. All these methodologies rely on the well-defined descriptions provided by the web services and the underlying semantic web languages.

Our system aims to discover dynamically appropriate candidate tools, using information extraction techniques on clinical questions guided by ontologies. So far no system that uses regular expressions for service discovery existed. Initial implementation and preliminary experiments has demonstrated technical feasibility and promising results of the proposed system.

The best results in both experiments (clinical questions) achieved using the forth methodology (EDAM Data Tags for tools input). This approach uses the annotated terms of the categories EDAM data and EDAM format which have been identified in the input part of the sentence and the query is focus only in the description of the inputs of the tools. The first clinical question could be answered using only one tool, by the EDAM_DATA tags to create the query. We identified manually 50 tools and most of them could answer this clinical question. This concludes to better results, because the probability for errors gets smaller as the number of retrieved tools is small. The second clinical question needed at least two tools to be answered. In both cases the last methodology identified appropriate tools to answer the initial clinical question with the minimum number of retrieved tools. Using only semantic types (SemTypes methodology) or only tags (Tags methodology) we achieved better result, with less and more specific tools than using the first query (Full text search). These results also showed us that the combination of biomedical and software domain ontologies is more effective, than in the use case of only one domain.

Taken altogether we are convinced that the recommended approach is technically feasible. We plan to expand the implementation with more regular expressions focus on the software and the clinical domain and also create more complex queries to the tools repository. Future enhancements will also take into account the ontology-based relationships of concepts and how there map to similar structure in the natural language. Furthermore we expect that soon enough the tools repository will host thousands of software descriptions making such a methodology mandatory for an effective and productive environment.

Acknowledgments. The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement N° 270089.

References

[1] Turing, A.M.: Computing machinery and intelligence. Mind 59(236), 433–460 (1950)
[2] Priyadharshini, G., Gunasri, R., Saravana Balaji, B.: A Survey on Semantic Web Service Discovery methods. International Journal of Computer Applications 82(11), 8–11 (2013)

[3] Json, J., et al.: EDAM: an ontology of bioinformatics operations, types of data and identifiers, topics and formats. Bioinformatics 29(10), 1325–1332 (2013)
[4] Bodenreider, O.: The unified medical language system (UMLS): integrating biomedical terminology. Nucleic Acids Research 32(1), D267–D270 (2004)
[5] Sager, N., Lyman, M., Bucknall, C., Nhan, N., Tick, L.J.: Natural language processing and the representation of clinical data. Journal of the American Medical Informatics Association 1(2), 142–160 (1994)
[6] Smiley, D., Pugh, D.E.: Apache Solr 3 Enterprise Search Server (2011)
[7] Aronson, A.R., Lang, F.-M.: An overview of MetaMap: historical perspective and recent advances. Journal of the American Medical Informatics Association 17(3), 229–236 (2010)
[8] Karttunen, L., Chanod, J.-P., Grefenstette, G., Schille, A.: Regular expressions for language engineering. Natural Language Engineering 2(4), 305–328 (1996)
[9] Henninger, S.: Using iterative refinement to find reusable software 11(5) (1994)
[10] Ritchie, G.D., Thanisch, P., Androutsopoulos, I.: Natural Language Interfaces to Databases - An Introduction. Natural Language Engineering 1(1), 29–81 (1995)
[11] Chapman, A., Elkiss, A., Jayapandian, M., Li, Y., Nandi, A., Jagadish, C.Y.H.V.: Making database systems usable. In: Proceedings of the 2007 ACM SIGMOD International Conference on Management of data (SIGMOD 2007), pp. 13–24. ACM, USA (2007)
[12] Jeong, M., Wang, Y.-Y., Hakkani-Tur, D., Heck, L., Tur, G.: Exploiting semantic web for unsupervised statistical natural language semantic parsing. In: Proceedings of Interspeech (2012)
[13] Moustakis, L., Potamias, V., Koumakis, G.: Web Services Automation. In: Oliveira, E.F., Tavares, A.J.V., Ferreira, L.G., Cruz-Cunha, M.M. (eds.) Handbook of Research on Social Dimensions of Semantic Technologies and Web Services. ch. 239. Information Science Reference, New York (2009)
[14] Kaufmann, E., Bernstein, A.: How Useful Are Natural Language Interfaces to the Semantic Web for Casual End-Users? In: Aberer, K., Choi, K.-S., Noy, N., Allemang, D., Lee, K.-I., Nixon, L.J.B., Golbeck, J., Mika, P., Maynard, D., Mizoguchi, R., Schreiber, G., Cudré-Mauroux, P. (eds.) ASWC 2007 and ISWC 2007. LNCS, vol. 4825, pp. 281–294. Springer, Heidelberg (2007)
[15] Sangers, J., Frasincar, F., Hogenboom, F., Hogenboom, A., Chepegin, V.: A linguistic approach for semantic Web service discovery (2012)
[16] Miguel, J., Rico, M., García-Sánchez, F., Béjar, R.M., Gómez, C.B.: GODO: Goal Driven Orchestration for semantic web services, Frankfurt, Alemania (2004)
[17] Aronson, A.R.: Effective mapping of biomedical text to the UMLS Metathesaurus: the Meta Map program. In: Proceedings of the AMIA Symposium, pp. 17–21. American Medical Informatics Association (2001)
[18] Smith, L., Rindflesch, T., Wilbur, J.W.: MedPost: a part of speech tagger for biomedical text. Bioinformatics 20(14), 2320–2321 (2004)
[19] Marcus, M.P., Marcinkiewicz, M.A., Santorini, B.: Building a large annotated corpus of English: the penn treebank. Computational Linguistics - Special issue on using large corpora: II 19(2), 313–330 (1993)

Automatic Generation of Questionnaires for Supporting Users during the Execution of Declarative Business Process Models

Andrés Jiménez-Ramírez[1], Irene Barba[1], Barbara Weber[2], and Carmelo Del Valle[1]

[1] University of Seville, Dpto. Lenguajes y Sistemas Informáticos, Spain
{ajramirez,irenebr,carmelo}@us.es
[2] University of Innsbruck, Department of Computer Science, Austria
barbara.weber@uibk.ac.at

Abstract. When designing an imperative business process (BP) model, analysts have to face many design requirements (e.g., managing uncertainty, optimizing conflicting objective functions). To facilitate such design, declarative BP models are increasingly used. However, how to execute a given declarative model can be quite challenging since there are typically several variants related to such model, each one presenting different degree of goodness. To support users working on declarative models while a high flexibility is maintained, we propose removing the worst variants from the source declarative model at design time while keeping the best variants. This way, the variants which are kept are narrowed down incrementally during run-time. For managing these variants during run-time we suggest to build upon configurable BP models. To configure such models, we additionally propose to automatically generate questionnaires. The results over a real case study are promising.

Keywords: Declarative Business Process Models, Configurable Business Process Models, Questionnaires.

1 Introduction

Business Process (BP) models are typically specified by hand using imperative languages like EPC or BPMN [3]. When designing an imperative BP model, analysts have to face many design requirements (e.g., dealing with activity attributes, managing uncertainty, dealing with relations between activities, considering the optimization of potentially conflicting objective functions, etc. [11]). To facilitate the human work involved in such design, to avoid failures, and to allow for a better optimization during the execution phase [7], declarative BP models are increasingly used allowing their users to specify what has to be done instead of how [1,10]. However, due to their flexible nature, there are frequently several variants related to a given declarative model, each one presenting specific values for different objective functions (e.g., overall completion time or profit). Therefore, the decision about how to execute this declarative model (i.e., selecting a variant that finally gets executed) can be quite challenging.

W. Abramowicz and A. Kokkinaki (Eds.): BIS 2014, LNBIP 176, pp. 146–158, 2014.

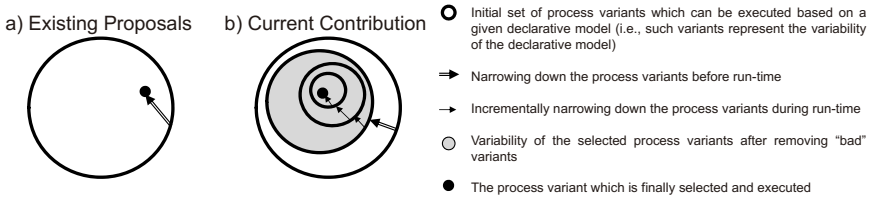


Fig. 1. Current contribution VS existing proposals

For supporting users working on declarative BP models, we proposed in previous works [2,1] an approach for generating an optimized execution plan (i.e., an optimized variant) from a given declarative BP model at design time (cf. Fig. 1 (a)). However, as a major disadvantage of such previous work, only one single variant is selected before starting the process execution which unnecessarily restricts the flexibility (cf. black dot in Fig. 1 (a)) [1,10], and hence, diminishes the advantages of using declarative models. In particular, if BPs are subject to uncertainty and conditions may change during the BP execution, it might turn out that the selected variant is not applicable and replanning might be required. To be better able to cope with such uncertainty, it is more suitable to defer the decisions of how the variant to be executed looks like to run-time. To be more specific, instead of narrowing down the selection to one single variant before run-time (cf. Fig. 1 (a)), it would be better that only *the worst* variants are removed while the *the best* variants are kept (cf. the outermost gray circle in Fig. 1 (b)). Thereby the goodness of a variant is measured by its values for given objective functions [10]. This way, the variants which are kept can be narrowed down incrementally during run-time at the last possible moment (i.e., gradually moving from the outermost inner circle to the back dot in Fig. 1 (b)).

To support users working on declarative BP models while a high flexibility is maintained, we propose unlike [2,1] to not select a single variant, but to keep the best variants before the BP enactment. For managing these variants during run-time, they can be automatically merged into a configurable BP (CBP) model (i.e., a modeling artifact that captures families of BP models in an integrated manner [16,15]) by using the techniques presented in [9]. Such a model then allows analysts to understand what these variants share, what their differences are, and why and how these differences occur.

To configure CBP models in such a way that domain experts incrementally reduce the number of process variants to be executed, we additionally propose to automatically generate questionnaires (i.e., sequences of questions each one created for configuring a part of a CBP model [12]). While the usage of questionnaires is not new [12,13], existing works require that analysts manually create the questionnaires, unlike the current proposal. In addition, such a configuration is done at configuration time (i.e., before process execution starts), and hence, unlike the current proposal and other proposals as aspect-oriented approaches [6], premature decisions may unnecessarily be taken.

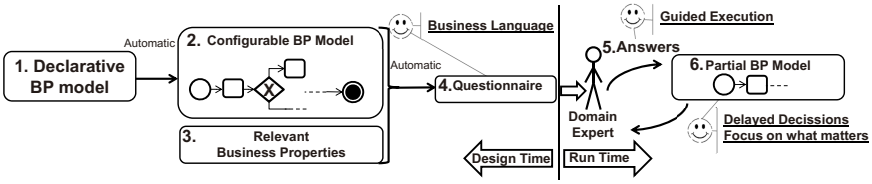


Fig. 2. Overview of our contribution

The first part of the current contribution (i.e., the generation of the best variants to be kept from a declarative BP model and the creation of a CBP model out of them) has been already presented in previous works [9,10]. However, this paper significantly extends [9,10] by: (1) Automatically generating the questionnaires for configuring the CBP model and (2) incrementally configuring the CBP model during run-time using the generated questionnaire.

As depicted in Fig. 2, in our proposal, a declarative BP model (cf. Fig. 2 (1)) is used as starting point. Then a CBP model is automatically generated out of it (cf. Fig. 2 (2)) by selecting the best variants as detailed in [9,10]. Then, using the generated CBP model together with a set of well-defined relevant business properties (i.e., properties that can be measured within each variant and which are understandable by the domain expert, cf. Fig. 2 (3)), we automatically generate a questionnaire without involving the analyst. Such a questionnaire consists of questions written in the business language (cf. Fig. 2 (4)). Thereafter, the domain expert interacts with the questionnaire to configure the CBP model herself during run-time. This way, the generated questionnaire allows to narrow down the variants of the CBP model in an incremental way during run-time, i.e., guiding the execution of the CBP model by answering the questionnaire (cf. Fig. 2 (5)). Therefore, the BP model is partially created (cf. Fig. 2 (6)) and thus, it is possible to execute already configured parts. In addition, as users often do not have an understanding of the overall process, they can focus only on the part of the CBP model to be configured, which may help them to take decisions.

Note that our approach is appropriate for managing scenarios which present certain requirements, i.e., scenarios which (1) are subject to changes (e.g., company best practices which change due to the customers feedback), (2) have a well-defined set of business properties which can be extracted for a variant (e.g., the property 'completion time' of a variant can be related to the 'opening and closing time' of the business), and (3) highly rely on domain expert's skills (i.e., decisions influence business performance) and thus, decisions can not be predefined. As an example of such a scenario, the suitability of the current proposal has been validated through a real scenario. Nevertheless, the proposed approach is not restricted to such scenarios, but can be applied over any CBP model.

This paper is organized as follows: Sect. 2 introduces backgrounds on related areas, Sect. 3 details the proposed method, Sect. 4 deals with the evaluation, and Sect. 5 includes some conclusions and future work.

2 Background

Declarative Models: Different paradigms for process modelling exist, e.g., imperative [3] and declarative [7]. Imperative process models are well structured representations which specify exactly how things have to be done by explicitly depicting all possible behavior. A declarative model, in turn, is a loosely-structured representation focused on what should be done restricting all forbidden behavior. Therefore, declarative models are commonly used for representing processes with high variability which can be executed in several ways (cf. Example 1). In the context of declarative models, a constraint-based model can be defined as a set of activities which can be executed following a given set of behavioural constraints (e.g., resource and control flow constraints).

Example 1. Figure 3 (a) shows a constraint-based BP model together with some valid and invalid traces[1]. In contrast, Fig. 3 (b) shows an imperative model where there is only one valid execution trace.

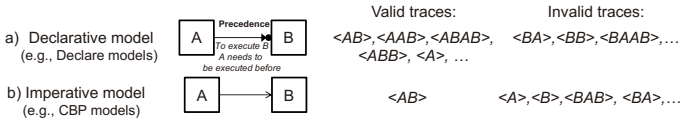


Fig. 3. Increased flexibility of declarative models versus imperative models

Due to their flexible nature, there are frequently different ways to execute a constraint-based model in such a way that all constraints are fulfilled. The different valid execution alternatives (i.e., variants), however, can vary significantly in how well different performance objective functions (e.g., benefit and time) can be achieved. For generating variants from a constraint-based model optimizing given objective functions, we applied planning & scheduling techniques in previous work [10]. Each variant which is generated can be represented as a BP graph (cf. Def. 1).

Definition 1. *A **BP Graph** $G = (gid, N, Pairs)$ is identified by gid and consists of a set of pairs of nodes $n \in N$, i.e., $Pairs$. Each pair denotes a direct edge between two nodes in the graph. A node $n \in N$ is a tuple $< nid, l, t >$ where nid is an unique identifier of a node in the graph, l is its label, and t is its type (e.g., activities, events, and gateways).*

Such definition of graph allows to represent a BP model in many different imperative BP languages [3], e.g., BPMN or EPC. As an example, the types of nodes (i.e., t) in BPMN language [3] are 'activity', 'event', or 'gateway'. A

[1] For the sake of clarity, traces represent sequences of activities, i.e., no parallelism is considered in the examples. Moreover, only completed events for activity executions are included in the trace representation.

node of type 'gateway' allows the labels (i.e., l) 'AND', 'OR', 'XOR', etc., while 'event' nodes allow 'start' and 'end' labels.

Configurable BP Models: Typically, different variants can be performed in scenarios which entail high variability. In most cases these variants share many commonalities, and hence, can be combined in a CBP model leading to a compact representation [15,16,12]. CBP models are typically created by hand (1) from scratch, (2) from an existing BP model by including possible adaptations [8], or (3) by merging some BP models related to the same or similar goals which already exist [15]. In the latter case, there exist approaches focused on automatically merging different BP models into a CBP model [14,15].

In a similar way to each variant can be represented as a BP graph (cf. Def. 1), CBP models can be represented as CBP graphs, which are defined as described in [15] (cf. Def. 2).

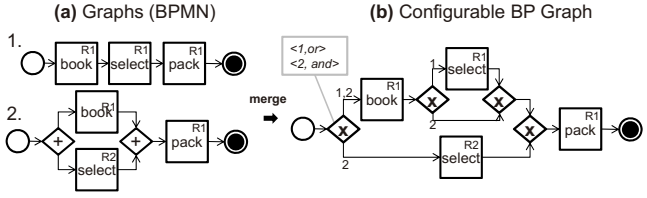


Fig. 4. Two BP graphs (a) are merged into a single configurable BP graph (b)

Definition 2. *A **Configurable BP Graph** $CG = (G, E2I, N2LI)$ consists of: (1) a BP graph, $G = (gid, N, Pairs)$ (cf. Def. 1), (2) a function $E2I$ that maps each edge $e \in Pairs$ to a set of BP graph identifiers (i.e., $E2I$ states which branches of CG belong to each source BP graph which is merged in CG), (3) a function, $N2LI$ that maps each node $n \in N$ to a set of pairs $< gpid, l >$ where gpid is a BP graph identifier and l is the label of the node n in the graph gpid (i.e., $N2LI$ states which nodes, with the corresponding label, belong to each source BP graph which is merged in CG).*

A CBP graph includes *configuration nodes* for those points where the BP graphs which are merged differ (cf. Example 2). Therefore, each branch and each node of the CBP graph can be related either to one or more BP graphs. As mentioned in Def. 2, to store these relations, each branch/node of the configurable BP graph includes identifiers related to the corresponding BP graph (i.e., $E2I$ function). In addition, the nodes of the CBP graph also store the associated label related to each identifier (i.e., $N2LI$ function).

Example 2. Figure 4 shows 2 BP graphs which are merged into a CBP graph [2]. As can be observed, the first gateway in Fig. 4(b) is a configurable node which

[2] As there is not ambiguity, some labels are not shown (i.e., they are the same as in the branch).

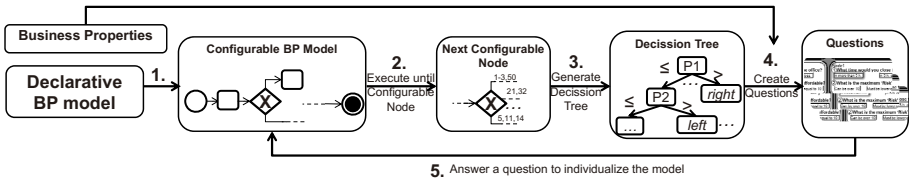


Fig. 5. Automatic generation of questionnaires for Individualizing a CBP model

corresponds to an 'OR' gateway in BP Graph 1 (it does not explicitly appear) and an 'AND' gateway in BP Graph 2.

Questionnaire Models: Questionnaire models [13] are typically created by the analysts to support the user during the configuration (i.e., individualization) of the CBP models. The main benefits of using them are: (1) they guide the user in such a way that choices are presented in a proper order and (2) they avoid invalid configurations which may lead to errors.

Typically, questionnaires are manually created by an analyst whereby each question is related to boolean *facts* which are associated to configuration *actions* [13]. Each time a question is answered, an action which configures a part of the CBP model is fired. The sequence of answers given to the different questions will individualize the CBP model in such a way that a single variant is selected before run-time to be executed.

Unlike previous approaches which deal with questionnaires, this work: (1) automatically creates the questionnaires (i.e., defining facts and actions are not longer needed) and (2) the questionnaires which are created are intended to individualize the CBP models during run-time (cf. Sect. 3).

3 Individualizing a CBP Model through the Automatic Generation of Questionnaires

In this section, our method for automatically generating questionnaires from a declarative model and its usage for supporting the user during the execution of such model is described (cf. Fig. 5). As a first step, a CBP model is generated out of the source declarative model (cf. Fig. 5 (1)) as detailed in [9,10]. Then, the BP execution starts and advances until a configurable node (cf. Def. 2) is found in the CBP model (cf. Sect. 3.1, Fig. 5 (2)). Thereafter, a decision tree related to such configurable node is created (cf. Sect. 3.2, Fig. 5 (3)) as an intermediate step for generating the questionnaire associated to this configurable node (cf. Sect. 3.3, Fig. 5 (4)). Whenever the user answers a questionnaire (i.e., a decision is taken, cf. Sect. 3.4, Fig. 5 (5)), the variants of the CBP model are narrowed down based on the answers given. This method is iteratively applied from step 2 to step 5 until no more individualization is needed (i.e., until only one single variant remains in the CBP model).

3.1 Executing the Configurable BP Model

As stated in Def. 2, all variants which are included in the CBP model are labeled (cf. Example 3).

Example 3. The running example of Fig. 6(a) comprises four BP models, each one labeled with an integer. Furthermore, a group of properties for each BP model is provided (cf. Fig. 6 (b) where time (T), benefit (B) and risk (R) properties are provided for each model). Such properties are related to the business language, e.g., T is related to the opening hours of the business. The CBP model associated with the BP models which are depicted in Fig. 6 (a) is shown in Fig. 6 (c). In this model, 4 different configurable nodes are depicted with a bold diamond. In the first configurable node, labeled as 1, two alternatives are possible. The *lower* branch comprises BP Model 4 (i.e., where activity A is not executed), and the *upper* branch comprises BP Models 1 to 3 (where activity A is executed).

The CBP model can be executed from the beginning until a configurable node appears, i.e., until a decision must be taken (cf. Fig. 5 (2)).

Note that the selection of a valid variant is guaranteed since we are building upon our previous work which generates valid variants. Merging them preserves these variants and the same happens with the decision trees.

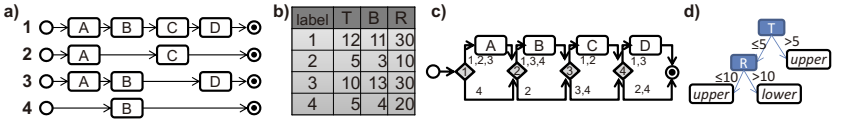


Fig. 6. a) 4 different BP models. (b) Properties of each BP model. (c) CBP model related to the BP models of (a). (d) Classification tree for node 1.

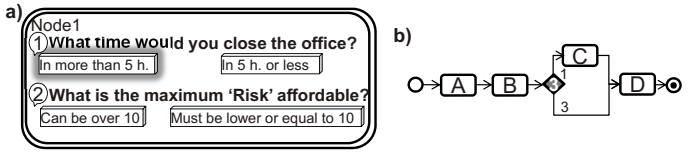


Fig. 7. (a) Questionnaire for Node 1. (b) The resulting configurable model after removing Variants 2 and 4.

3.2 Generating Decision Trees

When a configurable node is encountered we apply a method for generating a prediction system (i.e., a model that predicts the value of a target variable based on several input variables) [4] for predicting which outgoing branch corresponds to a given assignment of property values. Specifically, for each configurable node, a classification tree is created (cf. Fig. 5 (3)) using the property values of the variants as input variables and the outgoing branches as target variables (cf. Example 4).

Example 4. Figure 6(d) shows the classification tree which comes of using the CART algorithm [4] when using the table of Fig. 6(b) as input variables and the strings *lower* and *upper* as target variables. As can be seen, in the resulting classification tree, the variants for which $T > 5$ correspond to the *upper* branch. In contrast, the variants for which $T \leq 5$ correspond to the *upper* branch if $R \leq 10$, or to the *lower* branch otherwise.

3.3 Creating Questions

A set of questions is then created for each decision tree (cf. Fig. 5 (4)). To create such questions according to the business language, a set of well-defined business properties must be provided. This way, one question is automatically generated for each intermediate node of the tree. The possible answers for such question are the different labels which are written on the outgoing branches of this node. The text of the questions is automatically generated from the information of the provided business properties (cf. Example 5). As stated, these questionnaires are in charge of narrowing down the variants of the CBP model.

Example 5. A simple questionnaire related to the decision tree of Fig. 6(d) is shown in Fig. 7(a). Since this decision tree has two intermediate nodes (i.e., T and R), two questions are created. Moreover, since each node has two branches, each question has two options. Initially, only the question related to T is enabled. Considering that the well-defined business properties stated that T is related to the *closing time* of the office, the generated question would look like *What time would you close the office?*.[3] The second question has to be answered only if the user selects the second option of the first question (i.e., *In 5hrs. or less*) which is related to the branch $T \leq 5$ of the decision tree.

3.4 Incremental Configuration

Whenever a questionnaire is answered, the CBP model is narrowed down by removing the variants that do not belong to the edge selected in this configuration step. Thereafter, the proposed method continues at Step 2 (cf. Fig. 5) considering the narrowed CBP model and continuing the execution from the last executed activity.

Such method is repeated until only one variant remains in the CBP model, i.e., the configuration has finished (cf. Example 6).

Example 6. Supposing that the user selects the first answer of the first question of the questionnaire of Fig. 7(a) (i.e., *In more than 5hrs.*), Variants 2 and 4 are removed from the CBP model since they have a time property "≤ 5". This results in the CBP model of Fig. 7(b). Note that the second and forth configurable node

[3] Note that, the semantic of the generated questions highly depends on the information provided for the business properties. Such information can be used to make the questions more user-friendly. No depth details are given since it is out of the scope of this paper.

of Fig. 6(c) are not depicted in Fig. 7(b) since Variants 1 and 3 share the same outgoing branches for these nodes, i.e., the *upper* branch. However, the third configurable node requires selecting one of the two branches, and hence, a new questionnaire is generated.

4 Case Study

This section provides and empirical study for evaluating the suitability of the proposed approach. Specifically, the case study protocol proposed by [5] is followed to improve the rigor and validity of the study.

Background: In the context of the proposed approach, the *object of study* is the method for automatically creating questionnaires for CBP models and the method for incrementally configuring them (cf. Sect. 3). In such context, the *purpose of this study* is to evaluate the suitability of both methods regarding its feasibility and effectiveness when managing a real scenario.

Considering the object and purpose of this study, a main research question is defined: ($MQ1$) Is our method appropriate for individualizing CBP models during run-time? This question is further divided into 4 additional questions: ($AQ1$) Can the proposed method be used to generate questions for configurable nodes of different sizes (i.e., nodes with different number of branches)?, ($AQ2$) Are the generated questionnaires appropriate to be answered in a real environment (i.e., adequate number of questions)?, ($AQ3$) Is the business performance improved by using the proposed method?, and ($AQ4$) Is the proposed method preventing replanning (i.e., changing the variant which is being executed)?

Design: Two different designs are carried out in this study:

1. An *embedded* design considering Steps 2 and 3 of our approach (i.e., creating questionnaires) for addressing $AQ1$ and $AQ2$. For this, such steps are applied over a set of configurable nodes of different sizes. Such configurable nodes are part of different CBP models which are generated to represent some days of work in a business. In this design, we quantified for each configurable node: (1) the number of outgoing branches (cf. OB in Table 1), (2) the minimum, and (3) maximum number of questions which need to be answered for resolving the questionnaire associated to such node (cf. $\#mQ$ and $\#MQ$ respectively in Table 1).

   In addition, the business manager specified that answering more than 10 questions would be inefficient and thus, $AQ2$ can be answered as true if $\#MQ$ stays under 10 independently of the size of the configurable node.
2. An *holistic* design which regards the whole approach (cf. Sect. 3) is considered for addressing $AQ3$ an $AQ4$. Specifically, the current approach is applied over different CBP models each one presenting a different complexity (i.e., different number of activities). In this design, we quantified for each CBP model: (1) The number of activities of the CBP model (cf. $\#Acts$ in Table 2), (2) the number of questions which the user actually answers for individualizing the CBP model (cf. $\#Q$ in Table 2), (3) the increment of

profit which is obtained by using the current approach versus not using it (cf. $\Delta\$$ in Table 2), and (4) the percentage of cases in which replanning is avoided by using the current approach (cf. ΔR in Table 2).

Both designs are run on a Intel(R) Xeon(R) CPU E5530, 2.40GHz, 8GB memory, running Debian 6.0.3. After the application of such designs, the stored information is analyzed to answer the research questions.

Case Selection: For this case study, a real scenario which is detailed in a previous work (i.e., a beauty salon, cf. [10]) is selected. We consider this is a good and suitable case since it fulfills the following selection criteria: (1) it has been created for *an actual business*, (2) the business performance *highly relies on run-time decisions* (i.e., the knowledge of the domain expert has a great influence on the performance), and (3) the problem is *subject to variability*.

Case Study and Data Collection Procedure: A day of work in the beauty salon was modeled through a declarative specification using the language ConDec-R which was proposed in [10]. Considering the data related to each specific day of work (i.e., resource availability, services which are booked by the clients, etc.) a CBP model was generated for each day [10,9]. In addition, the salon manager provided a set of properties in form of functions (i.e., the well-defined properties written in the business language) which can be calculated from each variant which is included in the CBP model. For a period of 30 days, the following information was logged for each day through an application installed on the business:[4]

1. The CBP model which captures the different variants. As mentioned, only the best variants are kept when generating the CBP model.
2. The variant which was selected by the salon manager before starting the execution (i.e., before the first client arrived).
3. For each event that occurs during the day (e.g., when a client arrives, an activity starts or finishes), its time-stamp is recorded by the receptionist.

On the one hand, after the period of 30 days passed, for the *embedded* design, we gathered all the configurable nodes which appeared in the CBP models which were stored. Specifically, 259 configurable nodes were obtained and the current approach was applied to generate the questionnaire associated to each node. For each node, OB, $\#mQ$ and $\#MQ$ were stored. To analyze the behavior of the method against different complexities, the 259 configurable nodes were grouped considering OB. In particular 4 groups were considered: $OB \in [2, 5)$, $OB \in [5, 8)$, $OB \in [8, 11)$ and $OB \in [11, 14)$ (cf. Table. 1).

On the other hand, for the *holistic* design, the salon manager was supported by our tool. To be more precise, each time a configurable node appears (i.e., a decision needs to be taken), a questionnaire was prompted and she answered it. At the end of each day, the $\#Q$ and the selected variant (i.e., the result of the individualization) were stored. In addition, such variant was compared with

[4] The declarative model, the data which were used, and the properties which were provided can be downloaded from `http://regula.lsi.us.es/BIS/data.zip`

Table 1. Quantified variables for the embedded design

OB	$\#mQ$	$\#MQ$
$[2, 5)$	1.2	6.3
$[5, 8)$	1.1	5.0
$[8, 11)$	1.2	5.9
$[11, 14)$	1.1	6.1

Table 2. Quantified variables for the holistic design

$\#Acts$	$\#Q$	$\Delta\$$	ΔR
$(40, 60]$	3.1	141.5	70.0
$(60, 80]$	4.1	189.1	60.0
$(80, 100]$	7.6	219.4	66.7
$(100, 120]$	8.0	239.8	75.0

the variant selected before starting the execution and $\Delta\$$ was calculated and stored for each day. Furthermore, we checked if the variant which was selected before starting the execution could withstand the events logged for that day. In case it would not, we stored if replanning was avoided by using our approach, i.e., ΔR is stored.[5] The value for ΔR is calculated as the percentage of times that our approach avoided replanning against the total number of times that replanning was required. To analyze the behavior of the method against different complexities, the 30 CBP models (each one corresponding to a day of work) are grouped considering $\#Q$. In particular 4 groups are considered: $\#Q \in [40, 60)$, $\#Q \in [60, 80)$, $\#Q \in [80, 100)$, and $\#Q \in [100, 120)$ (cf. Table. 2).

Analysis and Interpretation: In order to answer $AQ1$ and $AQ2$, Table 1 is analyzed. The values of the columns $\#mQ$ and $\#MQ$ reveal that the number of questions that need to be answered for each node seems to be independent of the number of branches (cf. OB) of the related node. In addition, no errors were observed when generating the questionnaires and, thus, $AQ1$ can be answered as true. Furthermore, $\#MQ$ is lower than 10 (i.e., the number that the salon manager specified as maximum) and, thus, $AQ2$ can be answered as true.

In order to answer $AQ3$ and $AQ4$, Table 2 is analyzed. As expected, $\#Q$ increases as $\#Acts$ increases, which indicates that more effort is required by the domain expert to individualize more complex CBP models. Even though, $\#Q$ is lower than 10 in all the cases, meaning that our approach can efficiently deal with real problems. Moreover, $\Delta\$$ increases as $\#Acts$ increases, which highlights the benefits of using the proposed approach in real cases and thus, $AQ3$ can be answered as true. Regarding the values of ΔR, we can conclude that the number of times that the salon manager needs to change her initial plan due to unexpected events (e.g., a client arrives later than expected or a resource becomes unavailable) is drastically reduced (i.e., almost 43% in most complex cases). Therefore, $AQ4$ and consequently $MQ1$ can be answered as true.

Validity Evaluation: With relation to the *construct validity*, it has to be addressed in how far the measures which have been used are appropriate to address the research questions which have been planned. Firstly, the complexity of the

[5] Note that cases in which replanning becomes necessary may exist although run-time configuration is applied. In such situations, a new CBP model is created ensuring that all the included variants cover the given situation as discussed in [1,2].

problems which are considered is controlled by the number of branches of the configurable nodes and the number of activities of the CBP models in the embedded and the holistic design respectively. Although we consider that the beauty salon is a suitable business due to its complexity, different ways of varying this complexity can be applied to mitigate this threat, e.g., by changing the properties specified by the salon manager. Moreover, the duration of the logged data (i.e., 30 days) can be a threat. To the best of our knowledge there is no metric which states how long data must be logged to obtain a meaningful log. To mitigate this threat, longer durations can be considered to get more data, and therefore, to increase the probability of finding situations where the algorithm does not perform well.

Regarding the *internal validity*, the results concerning $\#MQ$ can be biased due to that the value for the upper bound of the number of questions to be answered to configure a specific configurable node (specified by the salon manager) represents a subjective point of view. This threat is difficult to eradicate. However, different business experts can be consulted to have different view points.

Finally, the *external validity* considers in how far the obtained results could be generalized to any business. This generalization is threatened by the fact that the beauty salon was the unique scenario which was studied. Other scenarios can be considered to replicate this study in order to mitigate this threat.

5 Conclusions and Future Work

To support users working on declarative BP models while a high flexibility is maintained, we propose a method which is based on removing the worst variants from the source declarative model at design time while keeping the best ones. This way, the variants which are kept are narrowed down incrementally during run-time. For this, we suggest to build upon configurable BP models. To enable configuring such models, we additionally propose to automatically generate questionnaires, unlike previous approaches. The results over a case of study are promising. As future work we plan to (1) improve the semantics of the questions which are created since they seem too artificial in some cases, (2) analyze more in depth the different classification algorithms for creating the decision trees, and (3) conduct experiments over other real scenarios for being able to generalize our results.

References

1. Barba, I., Del Valle, C., Weber, B., Jimenez-Ramirez, A.: Automatic generation of optimized business process models from constraint-based specifications. Int. J. Cooper. Inform. Syst. 22 (2013)
2. Barba, I., Weber, B., Del Valle, C., Jimenez-Ramirez, A.: User recommendations for the optimized execution of business processes. Data & Knowledge Engineering 86, 61–84 (2013)
3. Business Process Model and Notation (BPMN), Version 2.0. (2011), `http://www.omg.org/spec/BPMN/2.0/` (accessed June 1, 2011)

4. Breiman, L.: Classification and regression trees. The Wadsworth and Brooks-Cole statistics-probability series. Chapman & Hall (1984)
5. Brereton, P., Kitchenham, B., Budgen, D.: Using a protocol template for case study planning. In: Proceedings of EASE 2008. BCS-eWiC (2008)
6. Charfi, A., Müller, H., Mezini, M.: Aspect-oriented business process modeling with AO4BPMN. In: Kühne, T., Selic, B., Gervais, M.-P., Terrier, F. (eds.) ECMFA 2010. LNCS, vol. 6138, pp. 48–61. Springer, Heidelberg (2010)
7. Ferreira, H.M., Ferreira, D.R.: An integrated life cycle for workflow management based on learning and planning. Int. J. Cooper. Inform. Syst. 15(4), 485–505 (2006)
8. Gottschalk, F., van der Aalst, W.M.P., Jansen-Vullers, M.H., La Rosa, M.: Configurable workflow models. J. of Cooper. Inform. Syst. 17(2), 177–221 (2008)
9. Jimenez-Ramirez, A., Barba, I., Del Valle, C., Weber, B.: Generating multi-objective optimized configurable business process models. In: RCIS 2012, pp. 1–2 (2012)
10. Jiménez-Ramírez, A., Barba, I., del Valle, C., Weber, B.: Generating multi-objective optimized business process enactment plans. In: Salinesi, C., Norrie, M.C., Pastor, Ó. (eds.) CAiSE 2013. LNCS, vol. 7908, pp. 99–115. Springer, Heidelberg (2013)
11. Karim, A., Arif-Uz-Zaman, K.: A methodology for effective implementation of lean strategies and its performance evaluation in manufacturing organizations. Business Process Management Journal 19(1), 169–196 (2013)
12. Rosa, M.L., Dumas, M., ter Hofstede, A.H.M.: Modelling business process variability for design-time configuration. In: Handbook of Research on Business Process Modeling (2008)
13. Rosa, M., Aalst, W.P., Dumas, M., ter Hofstede, A.H.M.: Questionnaire-based variability modeling for system configuration. Software & Systems Modeling 8(2), 251–274 (2009)
14. La Rosa, M., Dumas, M., Uba, R., Dijkman, R.: Merging business process models. In: Meersman, R., Dillon, T.S., Herrero, P. (eds.) OTM 2010. LNCS, vol. 6426, pp. 96–113. Springer, Heidelberg (2010)
15. Rosa, M.L., Dumas, M., Uba, R., Dijkman, R.M.: Business process model merging: An approach to business process consolidation. ACM Transactions on Software Engineering and Methodology (TOSEM) (2012)
16. Rosemann, M., van der Aalst, W.M.P.: A configurable reference modelling language. Inform. Syst. 32, 1–23 (2007)

Complexity-Aware Software Process Management: A Case of Scrum in Network Organization

Leszek A. Maciaszek[1,2] and Lukasz D. Sienkiewicz[1]

[1] Wrocław University of Economics, Poland
[2] Macquarie University, Sydney, Australia
{Lukasz.Sienkiewicz,Leszek.Maciaszek}@ue.wroc.pl

Abstract. Software quality models and standards distinguish between product and process quality. Clearly, process quality determines product quality, yet surprisingly little research has been done on complexity-aware software process management. In this paper, we consider a software process as a system that (like a product) must minimize dependencies between system elements, and therefore minimize its complexity. We do so by applying holonic thinking to software process management and by adjusting traditional Scrum method for managing work in network organizations (where third party service providers may act outside of the Scrum process).

Keywords: Scrum, Complexity, Process Management, Network Organization, 3rd party services.

1 Introduction

Software systems are ever more complex because scientific and technological advances allow solving ever more complex problems and addressing ever more complex application domains. Software complexity is a by-product of problem complexity. As observed by Fred Brooks as early as in 1987 in his landmark 'silver bullets' paper [1], complexity is an "essence" and inherent difficulty of software production – a "werewolf" not amenable to silver bullets.

Together with three other invariants of software production noted by Brooks (conformity, changeability, and invisibility), software complexity underpins all efforts to achieve software quality. It follows that to build quality into a system, we have to manage complexity and we need to ensure that software does not become more complex than a problem it solves.

Software quality models and standards, such as SQuaRE [2], tend to concentrate on software product quality, but recognize that it is not possible to produce a quality product without having a quality process that defines lifecycle activities. Quality needs to be built into a product and to do so a quality process is required. This in turn implies that we need to be able to evaluate process quality by the same approach that we use to evaluate product quality, i.e. by measuring its cognitive and structural (internal) properties and functional properties (divided in SQuaRE into external properties and quality in use properties).

W. Abramowicz and A. Kokkinaki (Eds.): BIS 2014, LNBIP 176, pp. 159–171, 2014.

The complexity of software systems is in the wires – in the linkages and communication paths between software objects (components, classes, services, etc.). The "wires" create dependencies between objects that may be difficult to understand and manage (a software object A depends on an object B, if a change in B necessitates a change in A).

Complexity-aware software process management should lead to a system in which dependencies between software objects are minimized. A resulting system presents then the minimal (the best) cognitive complexity (attested by the effort required to understand the software) and the minimal structural complexity (attested by the effort required to maintain and evolve the software).

The realization that object dependencies are more important than the objects themselves places software systems on the holistic end of scientific investigation [3]. The resulting whole is more than the sum of its parts. This also places software systems firmly within the context of general systems theory. "Systems theory looks at the world in terms of the interrelatedness and interdependence of all phenomena, and in this framework an integrated whole whose properties cannot be reduced to those of its parts is called a system." [4].

This line of reasoning, when applied to natural (biological) systems, has led Arthur Koestler [5] to the notion of *holon* (from the Greek word: "holos" = whole and with the suffix "on" suggesting a part, as in neutron or proton). A holon (e.g. a system) is an object that is both a whole and a part, and which exhibits two opposite tendencies: an integrative tendency to function as part of the larger whole, and a self assertive tendency to preserve its individual autonomy.

Koestler argues that the structure and behavior of a natural system is a multi-leveled, stratified hierarchy of sub-wholes, where the sub-wholes form the nodes, and the branching lines symbolize channels of communication and control [6]. A stratified hierarchy of holons is called by Koestler a *holarchy* to distinguish it from a network, but also from a hierarchy. A biological holarchy seems to be a hint given by nature for how to develop and manage complex human-made systems.

The holarchy hypothesis explains the abstractions needed for managing complexity of a software product as well as a software process. In this paper, we discuss cognitive and structural properties of a Scrum [7] software process in a network organization in which third party service providers may be unaware of the Scrum process and the firms that make up a network organization are likely to be using disparate methods for their internal development projects.

We propose an adaptation of the Scrum method to suit complexity-aware software process management in Scrum environment in a network organization using third party internal and external services:

- We refer to the network organization and determine how the third party services and the Scrum are interrelated.
- We use the 3C-Collaboration model (i.e. Communication, Coordination, Cooperation) [8] to classify collaboration, with emphasis on the teamwork varieties between Scrum roles.
- We take a holonic view on the process and method of software development.

- We propose a Scrum-based/Scrumban [9] software development model that specifically includes some novel and excludes some conventional artifacts and rules.
- We propose a set of metrics (i.e. Key Performance Indicators) that help to control and coordinate proposed model.

The proposed model extends the model reported in [10]. It offers an innovative approach for systems development in network organizations that has evolved from the Scrum method and has been used and validated in practice.

2 Network Organization and Third Party Services in Terms of Software Development

The network organization is a collection of autonomous firms or units that behave as a single large entity (i.e. structure), using specific mechanisms to control and coordinate the entire project. The entities that make up that kind of organization are usually legally independent entities (separate companies). However, this is not the rule because some of them may be divisions within the company (sub-organization) that sell to outside customers, or they may be wholly owned subsidiaries providing the third party services to the entire network.

2.1 Third Party Services

For the determination of cognitive and structural complexity of a software process, we distinguish two types of third party services provided for software development in network organizations:

- Internal Services: time-consuming activities, which are important but additional to the entire project. Usually those are provided by subunits of a large company (e.g. internal UX expertise, internal testing, ICT support, etc.)
- External Services: all those issues that must not be covered by Internal Services and should be handled by other firms (e.g. authorized computer service, external testing service, translations, etc.).

We observe that companies that see their units as separate cost or profit centres (providers of internal services), may encourage the units to sell their services outside the company. The reason is that if the units have to operate within market, they will improve performance, better manage the prices, and of course earn money for the entire organization. The cooperation between services providers, usually establishes a long-term relationship between suppliers (i.e. providers of external services), who may then participate in planning sessions and influence the workload and schedule.

Both internal and external services may involve other third party services (internal or external) so the amount of dependencies, risk and uncertainties may be fast increasing. This in turn may adversely affect software development process by increase of complexity, deterioration of quality, changes in the scope, delays in the schedule, and all this may contribute to the project failure (more about third party services has been described in the paper [11]).

2.2 Communication, Coordination, Cooperation and Collaboration

We distinguish between the terms coordination, communication and cooperation. We use the 3C Collaboration model (i.e. 3C means: communication, coordination, cooperation) [8] to classify collaboration, with an emphasis on the group work implementation [11].

For a better understanding of the differences between these terms, we use the following definitions [11]:

- Communication (alias networking): a way how information is exchanged and used for sharing facts, human experience and practices, as well as feelings and rumors, for mutual benefit. It relates to how information is shared and how people understand each other. People can benefit from information exchange even without common goals and the common generation of value [12].
- Coordination**:** in addition to exchanging information, it achieves efficiency of aligning and altering activities and influences the efficiency of the achieved results. Each entity might use different resources and methods to create values or achieve goals at an individual level. There is a correlation between coordination and results, like in the traditional Scrum approach, where the Scrum Master is coordinating Scrum Team work by facilitating the whole Scrum process. We deal with coordination in the case of Scrum Master work. The main responsibility of the Scrum Master is to facilitate the work of Scrum Team, what in many cases means coordinating the Software Development Team and others in following the Scrum rules.
- Cooperation: is a process that involves sharing resources and adjusting activities for achieving compatible results, which provides efficiency for the cooperating entities. The work divides between the involved entities, thus aggregated value is the sum of outcomes generated by all participants in a nearly independent manner. A traditional Scrum software development method, based on the client-supplier relationship and the prescriptive roles is an example of the cooperative process, especially when we consider developing from scratch. This usually means that at the very beginning both the client and the supplier do not know what the result will be. Therefore, each participant performs its part of the work in cooperation with others (e.g. the Product Owner is defining the scope, the Software Development Team delivers a demo based on it, etc.). Thus, as an outcome we have a sum of measurable results. Of course, this does not preclude the existence of a common plan, which usually is roughly defined by a single entity (i.e. the customer) and requires co-working, especially when the results of one are delivered to the others. Collaboration: is the 4th item in 3C model seen as a collaboration triangle [11] where entities are sharing information, resources and responsibilities as well as designing and implementing activities, which are crucial to achieving common goals. We know that building mutual trust takes time, but effort and dedication as well as solving problems together, help in building commitment of all the involved entities and result in value creation as an effort of a team rather than individual contributions.

The 3C model is a guide to understanding teamwork in application software development. “While communicating, people negotiate and make decisions.

While coordinating themselves, they deal with conflicts and organize their activities in a manner that prevents loss of communication and of cooperation efforts. Cooperation is the joint operation of members of the group in a shared space, seeking to execute tasks, generating and manipulating cooperation objects. The need for renegotiating and for making decisions about non-expected situations that appear during cooperation may demand a new round of communication, which will require coordination to reorganize the tasks to be executed during cooperation."[13]

3 Agility in Network Organization

To highlight differences in the impact of Agile, Scrum and Engineering Practices in network organizations we take advantage of the "Three Level Framework" [14]. We propose the framework that takes the Scrum viewpoint and distinguishes three types of layers:

- Organization Level: all activities that are additional to Scrum (e.g. human resources, financial, capability, management, etc.), and identify the organization point of view (i.e. market, competitive advantage, priorities, products and services).
- Process Level: series of steps, rules and artifacts, which are used by the Scrum/Kanban team to produce the product or service. The goals of this level are developed from customer requirements (i.e. Sprint Planning) and benchmarking information (i.e. during Sprint Review/Retrospective Meeting).
- Job/Performer Level: all undertakings and instrumentation essential to achieving the goals of the process (e.g. code review, pair programming, continuous integration, etc.).

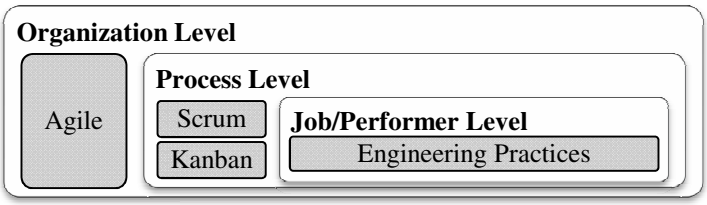


Fig. 1. Scrum and Engineering Practices in the context of Agile implementation in network organization

For the research presented here, we have assumed that network organization is following Agile principles and implements Scrum as a method for managing the software development team (Job/Performer Level in Fig. 1). Therefore, all Engineering Practices are considered separately - not covered by software development method (e.g. Scrum, Kanban, eXtreme Programming, etc.) but used as external or internal services.

In the following we do not focus on default core Scrum roles (i.e. Product Owner (PO), Software Development Team (SDT)), Scrum Master (SM), but only on additional roles that are essential for further consideration. In addition to core roles, we consider the groups of people in the capacities of managers and stakeholders.

For the sake of proper adaptation of Scrum to work with 3rd party services in network organization, we propose another core role, excluded from stakeholders group:

- Third Party Service Provider (S): organization or individual who provide your organization with specialized third party services (e.g. lawyers, accountants, coaches, consultants, translators, internal and external service providers, etc.).

From our point of view, this new role S is crucial to the success of the Scrum performed in network organizations. This role should be involved in the entire software development process.

4 Holonic Nature of Scrum

As explained in the Introduction section, the idea of holon was introduced by Arthur Koestler in [5]. He coined the term *"holon"* for those entities, which might be simultaneously a part and a whole and exhibit two opposite tendencies: an integrative tendency to exist as a part of complex system and a self assertive tendency to preserve its individuality.

Thirty years after Koestler's original idea, Ken Wilber generalized the idea of holons by highlighting its relative and conceptual nature. In [15], he considered that holon must have four basic characteristics [16]:

- Self-preservation: to maintain own structure, independently of the material that holon is made of.
- Self-adaptation (community): to adapt and link up with other superordinate holons, in order to react biologically, mechanically or intentionally to their stimuli.
- Self-transcendence: the holon has its own characteristics and qualities, which are new and emerging; new properties emerge in superordinate holons and create new classes of holons.
- Self-dissolution: the holons break up along the same vertical lines that they are formed.

Due to their nature, holons are connected to other holons in a typical vertical arborising structure known as a holarchy, which can be viewed as multilayer system architecture with tree-like structure.

We consider each member of a network organization) as a holon. It means that each member is a whole, if observed as a separate unit, and a part, if looked at as a member of larger organization. Therefore, the core and ancillary Scrum roles are interpreted as holons forming a holarchy built on a communication network designed between holons.

4.1 Holonic View of Scrum and Third Party Services in Network Organization

Considering network organization as a network of intercommunicating elements, we can easily show that the amount of communication paths, and therefore dependencies, grows exponentially with addition of new elements [17].

From the viewpoint of change management and control over system evolution, network is an inferior structure. We need some form of hierarchy with some aspects of superiority between elements. Holarchies seem to be the most suitable structures to manage complexity due to their special form of stratified hierarchy without traces of ranking between its elements (holons) and without cycles.

We propose a three layer holarchy (Fig. 2), where default core roles (PO, SM, ST) are placed in the first two layers and the new core role(S) is placed in the lowest layer. In this model, we skip ancillary roles of Managers and Stakeholders, because of low importance for further consideration.

All dependency relationships that represent requests for 3rd party services are downward. Superordinate layers depend on the sub-ordinate layers for third party services/vendors, however not vice-versa. The lower layers inform about its state changes or availability of created results by providing a feedback to the interested upper layers (or, in software engineering terms, by publishing an event to subscribed objects).

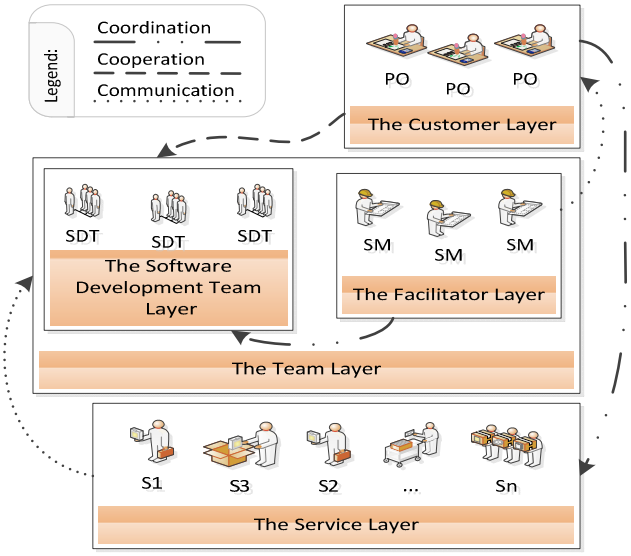


Fig. 2. Scrum-based model of information network in distributed projects

In our model most of the dependency relationships between layers are downward (e.g. cooperation, coordination), and upper communication is by providing the feedback, that helps in avoiding cycles of messages and makes communication more efficient.

The proposed model consists of three layers that refer to the core Scrum roles:

- The Customer Layer (TCL): with Product Owner/s (PO) as the main requester/s. This relation refers to coordination (mentioned in 2.2). The services are requested directly from this layer. However, results (i.e. communication) are delivered to The Software Development Team Layer (TSDL).
- The Team Layer (TTL): includes two sub-layers (i.e. The Software Development Teams Layer (TSDTL) and The Scrum Masters Layer (TSML)), where entities (holons) are Scrum Master/s (SM) and the Software Development Team/s (SDT). SDT receives results from S, requested by PO.
- The Services Layer (TSL): this layer represents third party services/vendors with third party services providers (S). It is possible that entities in this layer will have interconnectedness between each other (e.g. some S might request services from other S). It is also possible that TSL layer will have sub-layers, which will contain another set of above layers.

The same solution can be used in software development of distributed as well as co-located projects. The difference occurs in additional interconnectedness between entities from top layer (one or many PO's in case of complex products) and middle-layer (e.g. Scrum of Scrums Meeting may be represented as SM request service from another SM).

Our model supports building trust, multi-culture and massive design (what has been proved by the research results published in [10]). This is very important when technology and markets are changing very fast. The impact of S on the TTL layer results is essential and should not be omitted (e.g. estimates proposed by SDT during Sprint Planning should take into account S and dependencies associated with delivering results of third party services - delays in layer S affect results from layer SDT).

4.2 Scrum Artifacts

In this paper we assume that the reader is familiar with Scrum [7], therefore we describe only those Scrum artifacts that we propose to change to work better in network organizations:

- Report Meeting instead of Sprint Review Meeting: because regular Sprint Planning sessions involve many resources (i.e. a lot of participants), we propose to limit participants only to representatives of the customer and the team. In our opinion that kind of meeting should be held more frequently than Sprint Review Meeting (e.g. every week) in order to improve information flow between the customer and the team.
- Planning on demand instead of Sprint Planning Meeting: because we skipped time-estimations we also propose to limit number of Sprint Planning Meetings and hold them only if really needed (i.e. when the customer needs help from the team, because is not able to prioritize Product Backlog without an additional Team's expertise).

- Task-feasibility instead of time-estimation: we ignore formal time-estimates and try to commit only those User Stories that we are able to deliver before next demo session. Through this change, we want to limit commitment, which we are not able to provide, what seems to be natural if we want to organize the product backlog and do not practice Sprint Planning meetings.

The proposed changes arise from the fact that we adopted a holonic view that reduces the dependencies between the layers (e.g. no dependency cycles, stable workload during the Sprint, etc.). In our approach, the feedback notifications from S to PO must go through TTL, thus ensuring that TSDTL and TSML are fully involved in providing deliveries. For instance, the SDT will not be able to deliver implementation of a new wizard until they get all required translated texts from translators (i.e. S), so this implies that SDT and SM must keep an eye on S and their deliveries.

4.3 Key Performance Indicators

Within original Scrum we use only one metric (i.e. indicator). This is the time-estimate of the amount of remaining work that needs to be done versus amount of User Stories or Tasks that are set as "done" in Sprint Backlog [18]. We propose to use the following KPI's (i.e. Key Performance Indicators) that help better control software development in network organization:

- Reliability: to measure if the team is delivering what they said they will. We compare the difference between the amount of committed Story Points (ci) and delivered Story Points (di), as shown in (1). The values might be presented as the percent of reliability calculated per Sprint (Ri).

$$R_i = c_i / d_i * 100\% . \quad (1)$$

- Productivity: to measure project velocity. We measure amount of fixed bugs (bi) and newly implemented requirements (si), as shown in (2). The value of productivity (Pi) should be calculated after each Sprint.

$$P_i = b_i + s_i . \quad (2)$$

- Effectiveness: to measure effectiveness of testing service. The measure includes the amount of defects delivered to the customer. Based on this KPI we can calculate the effectiveness of internal testing service (as shown in (3)), by measuring the ratio between all found defects and those found by external S providing complementary testing. This shows effectiveness (Ei) of software development team and testing services.

$$E_i = (a_i – e_i) / a_i * 100 \% \quad (3)$$

The introduced KPI's are crucial to maintaining customer satisfaction. The required data should be collected at the end of each Sprint. We would like to point out that the

same KPI's can be measured for (S) and their findings can be used by the team for increasing customer satisfaction and for coordinating and controlling workload status.

In a large industrial field study reported in [10] we demonstrated an improved satisfaction of customers and other stakeholders when using our modified Scrum approach. Although in that field study we did not directly refer to the SQuaRE quality characteristics [2], it is obvious that the study findings as well as advantages of our holonic model and associated key performance indicators can be interpreted within the context of that standard. The SQuaRE quality characteristics amenable to such interpretations include quality-in-use characteristics (in particular satisfaction) and system/software product quality characteristics (in particular maintainability). These and other quality characteristics can improve by the virtue of improved process quality of our Scrum model.

5 Related Work

This paper extends the Scrum-based model published in [10] – and by addressing a software process – it provides an orthogonal viewpoint on complexity-aware architecture-centric software product management reported among others in [19], [20], [21].

Software process quality as sine-qua-non of software product quality has been uniformly acknowledged in the literature, including the SQuaRE standard [2]. A recent paper on evaluation and measurement of software process improvement offers a related systematic literature review [22]. The authors note that product complexity is a frequently measured attribute, but do not identify process complexity as a measurable entity. The fact that dependencies between software product objects determine software complexity has also been uniformly acknowledged [21] and supported by a variety of metrics [23] However, we do not know of research that would identify dependencies between software process objects in order to offer related complexity metrics.

Similarly, the holon abstraction has been studied to offer reference architectures to manage complexity, in particular in manufacturing systems [24]. Attempts have also been made to use the holonic abstraction for business process modeling [25]. However, again, we do not know of research that would apply the holon abstraction to a software process in order to offer a dependency-minimizing holarchy and facilitate process complexity management.

We can assume, on the basis of the research results and experience [7], [26], that the best choice for almost all kinds of software development projects, starting from scratch and executed in continuously changing environment, is an adaptive and flexible life cycle model (i.e. Agile) and a strongly prescriptive method (e.g. Scrum, eXtreme Programming, etc.) [7], [26]. We agree with Scrum advocates that using time-boxed delivery cycles (i.e. Sprints), visualization of the project scope (i.e. Product Backlog), prescribed roles (e.g. Scrum Master, Product Owner, Scrum Team), essential meetings (e.g. Sprint Retrospective, Sprint Review, Daily Meetings), and following Scrum rules is necessary for project's success.

Cellary and Picard presented in [27] the way to achieve agility and pro-activity by introducing the model of Collaborative network organizations in its two forms: Virtual Organizations (VO) and Virtual Organization Breeding Environments (VOBE). They presented idea of public administration "playing a role of Virtual Organization customer on the one hand, and a Virtual Organization member on the other hand" [27]. This publication was for us a stimulus for reflection about third party service providers (i.e. S role) as an entity that might be simultaneously a part and a hole in network organizations.

Fuks et al. [13] and Lucena et al. [8] introduce an approach based on the 3C (i.e. Cooperation, Communication, Coordination) model to the development of collaborative systems. The authors studied the 3C model "by means of a detailed analysis of each of its three elements, followed by a case study of learningware application and the methodology of a web-based course, both designed based on this model". From our point of view, the most interesting is the 3C collaboration model instantiated for teamwork. This interpretation of collaboration and its iterative nature seems to be adequate to our understanding of the dependencies between Scrum roles and other stakeholders (e.g. The participants obtain feedback from their action and feedthrough from the action of their companions by means of awareness information related to the interaction among participants" [13].

Mella studied the holonic perspective in organization and management. In [16] he examined six different examples of holonic networks in terms of manufacturing systems. This paper offered an inspiration to consider the network organization as holonic network, seen as "comprised of autonomous firms that are variously located—characterized by different roles and different operations and connected through a holonic network, real or virtual, often oriented, in order to achieve a common objective through the sharing of resources, information, and necessary competencies" [16].

Dirk S. Hovorka and Kai R. Larsen in [28] present the study that examined the influence of network organization environment on the ability to develop agile adoption practices. They use exploratory case study design to "investigate the interactions between network structure, social information processing, organizational similarity (homophily), and absorptive capacity during the adoption of a large-scale IT system in two network organization environments" [28]. They propose Agile Adoption Practice Model (i.e. APM) that proposes interactions within the inter-organizational network that enable Agile adoption practices. We adopted a more detailed approach, and instead of treating Agile life cycle model as a set of good practices, we proposed a more detailed analysis of one selected method of software development (i.e. Scrum) and propose Scrum-based model that suits better network Organizations.

6 Conclusion

We agree with the Scrum advocates that following the Agile Manifesto principles is possible only because the Scrum defines precisely the essential roles, principles and artifacts. This makes the method very prescriptive for managing software development. Firstly, in addition to the traditional Scrum, we propose to add a new role (i.e. third party service provider – *S*) and some extra rules for adapting the Scrum

and third party services to the network organization. Secondly, by referring to the network organization and third party services, we determine how they interrelate (i.e. Collaborate [8]) with the Scrum and Agile environment.

In this paper, we proposed a holonic view based on which we adapted Scrum for 3rd part services and the network organization. We developed a new service layer in the holonic structure and recommended new Scrum principles. In an industrial case study reported in [10], we demonstrated the advantages of our model and method.

We believe that big differences in the level of satisfaction of the customer using our approach were caused by very prescriptive way of working with the pure Scrum.

There is no easy way to adapt Scrum software development method to work in network organization; however, we believe that presented results will serve to advance research and help in finding the best solution.

In this paper, we have used the 3C model to distinguish teamwork types between Scrum "players". By extending our previous research reported in [10], we highlighted the differences between teamwork within entities taking part in application software development. The overwhelming concern was minimization of dependencies between holons of the software process in order to better manage cognitive and structural complexity of the process.

References

1. Brooks, F.: No Silver Bullet: Essence and Accidents of Software Engineering. IEEE Software 4, 10–19 (1987)
2. ISO/IEC2510: Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models (2011)
3. Maciaszek, L.: Modeling and Engineering Adaptive Complex Systems, Challenges in Conceptual Modeling. In: Grundy, J., Hartmann, S., Laender, L., Maciaszek, L., Roddick, J. (eds.) Proc. Tutorials, Posters, Panels and Industrial Contributions to the 26th International Conference on Conceptual Modeling - ER 2007, CRPIT, no. 83, pp. 31–38. ACS (2007)
4. Capra, F.: The Turning Point. Science Society, and the Rising Culture, p. 27. Flamingo, USA (1982)
5. Koestler, A.: The Ghost in the Machine. Penguin Group, England (1967)
6. Koestler, A.: Bricks to Babel, Random House (1980)
7. Schwaber, K., Beedle, M.: Agile Software Development with Scrum. Prentice-Hall, Upper Saddle River (2002)
8. Lucena, C.J.P., Fuks, H., Raposo, A., Gerosa, M.A., Pimentel, M.: The 3C Collaboration Model. In: Kock, N. (ed.) Encyclopaedia of E-Collaboration, pp. 637–644. IGI Global, Texas (2008)
9. Sienkiewicz, L.: Scrumban – the Kanban as an Addition to Scrum Software Development Method in a Network Organization. Business Informatics 2(24), 73–81 (2012)
10. Sienkiewicz, L., Maciaszek, L.: Adapting Scrum for Third Party Services and Network Organizations. In: Proceedings of the Federated Conference on Computer Science and Information Systems, pp. 329–336. IEEE Xplore DL (2011)
11. Sienkiewicz, L.: Collaboration Between the Scrum and Third Party Services in the Network Organization. Business Informatics 23(1), 59–66 (2012)

12. Camarinha-Matos, L., Afsarmanesh, H., Galeano, N., Molina, A.: Collaborative Networked Organizations – Concepts and Practice in Manufacturing Enterprises. Computers & Industrial Engineering 57(1), 46–60 (2009)
13. Fuks, H., Raposo, A.B., Gerosa, M.A., Lucena, C.J.P.: Applying the 3C Model to Groupware Development. International Journal of Cooperative Information Systems 14(2-3), 99–328 (2005)
14. Rummler, G., Brache, A.: Improving Performance - How to Manage the White Space in the Organization Chart, 2nd edn. Jossey Bass Inc., USA (1995)
15. Wilber, K.: A Brief History of Everything. Shambhala Publications, Massachusetts (2000)
16. Mella, P.: The holonic Perspective in Management and Manufacturing. International Management Review 5(1), 19–30 (2009)
17. Maciaszek, L.: Architecture-Centric Software Quality Management. In: Cordeiro, J., Hammoudi, S., Filipe, J. (eds.) Web Information Systems and Technologies. LNBIP, vol. 18, pp. 11–26. Springer, Heidelberg (2009)
18. Zabkar, N., Mahnic, V.: Using COBIT Indicators for Measuring Scrum-based Software Development. WSEAS Transactions on Computers 7(10), 1605–1617 (2008)
19. Maciaszek, L., Liong, B.: Practical Software Engineering. A Case-Study Approach, p. 864. Addison-Wesley (2005)
20. Maciaszek, L.: An Investigation of Software holons - The 'adHOCS' Approach. Argumenta Oeconomica 1-2(19), 1–40 (2007)
21. Sangal, N., Jordan, E., Sinha, V., Jackson, D.: Using Dependency Models to Manage Complex Software Architecture. In: OOPSLA 2005, pp. 167–176. ACM (2005)
22. Unterkalmsteiner, M., Gorschek, T., Cheng, K., Pemadi, R., Feldt, R.: Evaluation and Measurement of Software Process Improvement – A Systematic Literature Review. IEEE Trans. on Software Engineering 38(2), 398–424 (2012)
23. Perepletchikov, M., Ryan, C.: Controlled Experiment for Evaluating the Impact of Coupling on the Maintainability of Service-Oriented Software. IEEE Trans. on Soft. Eng. 37(4), 449–465 (2011)
24. Van Brussel, H.K., Wyns, J., Valckenaers, P., Bongaerts, L., Peeters, P.: Reference Architecture for Holonic Manufacturing Systems: PROSA. Computers In Industry 37(3), 255–276 (1998)
25. Clegg, B.: Building a Holarchy Using Busi-ness Process-Oriented Holonic (PrOH) Modeling. IEEE Trans. on Systems, Man and Cybernetics - Part A: Systems and Humans 37(1), 23–40 (2007)
26. Cockburn, A.: Selecting a Project's Methodology. IEEE Software 4(17), 64–71 (2000)
27. Cellary, W., Picard, W.: Agile and Pro-Active Public Administration as Collaborative Networked Organization. In: International Conference on Theory and Practice of Electronics Governance ICEGOV 2010. ACM, New York (2010)
28. Hovorka, D., Larsen, K.: Enabling Agile Adoption Practices Through Network Organization. European Journal of Information Systems - Including a Special Section on Business Agility and Diffusion of Information Technology 15(2), 159–168 (2006)

LCBM: Statistics-Based Parallel Collaborative Filtering*

Fabio Petroni[1], Leonardo Querzoni[1], Roberto Beraldi[1], and Mario Paolucci[2]

[1] Department of Computer Control and Management Engineering Antonio Ruberti, Sapienza University of Rome
{petroni,querzoni,beraldi}@dis.uniroma1.it
[2] Institute of Cognitive Sciences and Technologies, CNR, Italy
mario.paolucci@istc.cnr.it

Abstract. In the last ten years, *recommendation systems* evolved from novelties to powerful business tools, deeply changing the internet industry. Collaborative Filtering (CF) represents today's a widely adopted strategy to build recommendation engines. The most advanced CF techniques (i.e. those based on matrix factorization) provide high quality results, but may incur prohibitive computational costs when applied to very large data sets. In this paper we present Linear Classifier of Beta distributions Means (LCBM), a novel collaborative filtering algorithm for binary ratings that is (i) inherently parallelizable and (ii) provides results whose quality is on-par with state-of-the-art solutions (iii) at a fraction of the computational cost.

Keywords: Collaborative Filtering, Big Data, Personalization, Recommendation Systems.

1 Introduction

Most of todays internet businesses deeply root their success in the ability to provide users with strongly personalized experiences. This trend, pioneered by e-commerce companies like Amazon [1], has spread in the last years to possibly every kind of internet-based industries. As of today, successful players like Pandora or StumbleUpon provide user personalized access to services like a core business, rather than an add-on feature.

The fuel used by these companies to feed their recommendation engines and build personalized user experiences is constituted by huge amounts of user-provided data (ratings, feedback, purchases, comments, clicks, etc.) collected through their web systems or on social platforms. For instance, the Twitter micro-blogging service has surpassed 200 million active users, generating more than 500 million tweets (micro-blog posts) per day at rates that recently (Aug 2013) peaked at 143199 tweets per second [2]. The amount of data available

* This work has been partially supported by the TENACE PRIN Project (n. 20103P34XC) funded by the Italian Ministry of Education, University and Research.

W. Abramowicz and A. Kokkinaki (Eds.): BIS 2014, LNBIP 176, pp. 172–184, 2014.

to be fed to a recommendation engine is a key factor for its effectiveness [3]. A further key factor in this context is represented by timeliness: the ability to timely provide users with recommendations that fit their preferences constitutes a potentially enormous business advantage [4].

A widely adopted approach to build recommendation engines able to cope with these two requirements is represented by *Collaborative filtering* (CF) algorithms. The essence of CF lies in analyzing the known preferences of a group of users to make predictions about the unknown preferences of other users. Research efforts spent in the last ten years on this topic yield several solutions [5,6,7,8] that, as of today, provide accurate rating predictions, but may incur prohibitive computational costs and large time-to-prediction intervals when applied on large data sets. This lack of efficiency is going to quickly limit the applicability of these solutions at the current rates of data production growth, and this motivates the need for further research in this field.

In this paper we introduce *Linear Classifier of Beta distributions Means* (LCBM), a novel algorithm for collaborative filtering designed to work in systems with binary ratings. The algorithm uses ratings collected on each item (i.e. products, news, tweets, movies, etc) to infer a probability density function shaped as a Beta distribution; this function characterizes the probability of observing positive or negative ratings for the item. A linear classifier is then used to build user profiles that capture the aptitude of each user to rate items positively or negatively. These profiles are leveraged to predict ratings users would express on items they did not rate. Our algorithm is able to provide predictions whose quality is on-par with current state-of-the-art solutions (based on matrix factorization techniques), but in shorter time and using less computational resources (memory occupation). Moreover, it is inherently parallelizable. Its performance has been extensively assessed through an experimental evaluation based on well-known public datasets (MovieLens and Netflix) and compared with those offered by open source implementations of state-of-the-art solutions.

The rest of this paper is organized as follows: Section 2 presents related works; Section 3 defines the system model and states the problem; Section 4 presents our solution, evaluated in Section 5; finally, Section 6 concludes the paper.

2 Related Work

Collaborative Filtering (CF) is a thriving subfield of machine learning, and several surveys expose the achievements in this fields [9,10]. CF solutions in the literature are often divided in two groups: *memory-based* and *model-based* [11].

Memory-based methods [12,13] are used in a lot of real-world systems because of their simple design and implementation. However, they impose several scalability limitations that make their use impractical when dealing with large amounts of data. The slope one algorithms [14] were proposed to make faster prediction than memory-based algorithms, but they were unable to overcome the scalability issues of the latter.

Model-based approaches have been investigated to overcome the shortcomings of memory-based algorithms. The most successful Model-based techniques are by far those based on low-dimensional factor models, as the Netflix Prize (www.netflixprize.com) established, in particular those based on *matrix factorization* (MF) [15,16,5,6]. These methods aims at obtaining two lower rank matrices P and Q, for users and items respectively, from the global matrix of ratings R, with minimal loss of information. The most popular MF solutions are *Alternating Least Squares* (**ALS**) and *Stochastic Gradient Descent* (**SGD**). Both algorithms need several passes through the set of ratings to achieve this goal. ALS [5] alternates between keeping P and Q fixed. The idea is that, although both these values are unknown, when the item vectors are fixed, the system can recompute the user vectors by solving a *least-squares* problem (that can be solved optimally), and vice versa. SGD [6] works by taking steps proportional to the negative of the gradient of the error function. The term *stochastic* means that P and Q are updated, at each iteration, for each given training case by a small step, toward the average gradient descent. Some recent works aimed at increasing the scalability of current MF solutions [7,17,8], however the asymptotic cost of these techniques makes it difficult to fit the timeliness requirements of real-world applications, especially when applied on large data sets. Furthermore, each update leads to non-local changes (e.g. for each observation the user vector increment in SGD is proportional to the item vector, and vice versa) which increase the difficulty (i.e. the communication costs) of distributed implementations.

The binary-rating scenario we consider in this work can be considered as a special case of the more general multi dimensional rating scenario. However it is worth noticing that it fundamentally differs from *one-class collaborative filtering* [18] where only positive feedback are assumed to be available while negative feedback are treated as absent. Contrarily, in our work negative feedback is always considered at the same level of importance as positive feedback, but with an opposite meaning.

Some earlier works on collaborative filtering [19,20] and reputation [21] adopted the same statistical method (i.e. Beta distribution) to combine feedback. Ungar and Foster [19] proposed a clustering CF approach in which the connection probabilities between user and item clusters are given by a Beta distribution. The solution is computationally expensive, as Gibbs sampling is used for model fitting. Wang et al. [20] applied information retrieval theory to build probabilistic relevance CF models from implicit preferences (e.g. frequency count). They use the Beta distribution to model the probability of presence or absence of items in user profiles.

3 System Model and Problem Definition

We consider a system constituted by $U = (u_1, \cdots, u_N)$ users and $I = (i_1, \cdots, i_M)$ items. Items represent a general abstraction that can be case by case instantiated as news, tweets, shopping items, movies, songs, etc. Users can rate items with values from a predefined range. Rating values X can be expressed in several

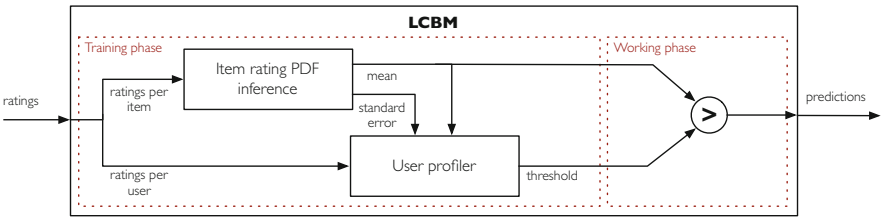


Fig. 1. LCBM: algorithm block diagram

different ways (depending on the specific system), however, in this paper we will consider only binary ratings, thus $X = \{-1, 1\}$ where the two values can be considered as corresponding to *KO* and *OK* ratings respectively.

By collecting user ratings it is possible to build a $N \times M$ rating matrix that is usually a sparse matrix as each user rates a small subset of the available items. The goal of a *collaborative filtering* system is to predict missing entries in this matrix using the known ratings.

4 The LCBM Algorithm

This section introduces the LCBM algorithm for collaborative filtering and analyzes its asymptotic behavior. First it describes the general structure of the algorithm and its internal functional blocks detailing their interactions; then the blocks are described in the following subsections.

4.1 Algorithm Structure

Our solution departs from existing approaches to CF by considering items as elements whose tendency to be rated positively/negatively can be statistically characterized using an appropriate probability density function. Moreover, it also considers users as entities with different tastes that rate the same items using different criteria and that must thus be profiled. Information on items and users represents the basic knowledge needed to predict future user's ratings. LCBM is a two-stage algorithm constituted by a training phase, where the model is built, and a working phase, where the model is used to make predictions. Figure 1 shows a block diagram of LCBM that highlights its two-stage structure, its inputs and outputs, its main functional blocks and the interactions among them.

Training phase in this first phase collected ratings are fed to both an *Item rating PDF inference* block and a *User profiler* block. In the former case ratings are grouped by item and the block performs statistical operations on them to infer for each item the probability density function (PDF) of positive/negative rating ratios. Each inferred PDF is described by two measures: the *mean* and the *standard error*. In the latter case ratings are grouped by user and the block

uses them to profile each user's rating behavior. It is important to note that in order to build accurate profiles this block is also fed with the data produced by the item's rating PDF inference block. The output of this block for each user is a single threshold value in the $[0, 1]$ range. The PDF mean values and the user thresholds represent the final output of this phase.

Working phase the second phase is in charge of producing the rating predictions. For each couple $(u, i), u \in U, i \in I$ such that the user u has not rated the item i a comparator is used to check the PDF mean value for that item against the user threshold and predict if that user will express a positive or negative rating.

It is important to notice that, while the flow of data between blocks in the algorithm architecture forces a sequential execution, operations performed within each block can be easily parallelized favoring a scalable implementation of the algorithm. Currently, we realized a prototype implementation of LCBM where the two blocks are implemented through multithreaded concurrent processes. More efficient and scalable implementations are part of our future work.

4.2 Item Rating PDF Inference

Items are profiled through a PDF of a single *random variable*, that represents the relative frequency of positive votes that the item will obtain in the future, given the observed ratings. We use the Beta distribution, a continuous family of probability functions indexed by two parameters α and β, to model this PDF. Given a number of received ratings, the unknown relative frequency of OKs an item will receive in the future has a probability distribution expressed by a Beta function with parameters α and β set to the number of OKs and KOs incremented by one respectively: $\alpha = OK + 1$ and $\beta = KO + 1$.

The profile of an item consists of two values: a measure of the beta distribution central tendency, the mean ($MEAN$), and a measure of the distribution variability, the standard error (SE):

$$MEAN = \frac{\alpha}{\alpha + \beta} = \frac{OK + 1}{OK + KO + 2} \tag{1}$$

$$SE = \frac{1}{OK + KO + 2} \sqrt{\frac{(OK + 1)(KO + 1)}{(OK + KO)(OK + KO + 3)}} \tag{2}$$

The $MEAN$ can be interpreted as the expected value for the relative frequency of OK votes that the item will obtain in the future. The SE is an estimate of the standard deviation of the $MEAN$. This value is important to indicate the reliability of an estimation. Intuitively, the more representative is the subset of voters, the lower the SE and the more accurate the $MEAN$ estimation.

Figure 2 shows the inferred PDF for an item that received so far 8 positive votes and 3 negatives. This curve expresses the probability that the item will receive a relative fraction of x positive ratings in the future. The mean of the distribution is 0.7. This can be interpreted as the expected value for x. For instance, the system expects that 7 of the next 10 ratings for the item will

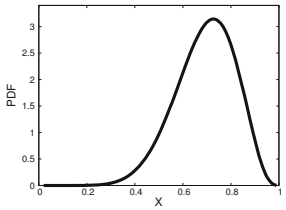


Fig. 2. Item profiling. Beta function after 8 OK and 3 KO

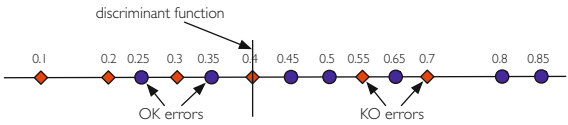


Fig. 3. User profiling. Linear classification in one dimension. OK represented by blue circles and KO by red diamonds. In this example $QT = 0.4$.

be positive. The standard error of the distribution is roughly 0.04. Using the Chebyshev's inequality [22], the SE can be interpreted as saying that the system expects that in the next 100 ratings for the item between 62 and 78 will be positive, with probability bigger than 0.75.

4.3 User Profiler

The goal of the *User profiler* is to identify a single value for each user using the votes that user expressed. We call this value the *quality threshold* (QT) of the user. Users profiling starts after the items profiling procedure. Therefore in this phase $MEAN$ and SE values are already defined for each item.

The algorithm uses a sorted data structure to collect all user's ratings. Without loss of generality, let this structure be a sorted set of points. Every rating is represented by a unique point p composed by two attributes: a key $p.key$, that gives the point's rank in the order, and a boolean value $p.value$ containing the rating. Each key lies on a $[0, 1]$ scale and its value is determined by the item's PDF. In particular, we adopt a *worst case estimation* approach: if the rating is positive (OK) the key is obtained by summing $2SE$ to the $MEAN$ of the item profile, if negative (KO) by subtracting $2SE$ from the $MEAN$.

A simple *linear classifier* is then used to find a *discriminant function* (i.e. a point p^*) for each user that separates the data with a minimal number of errors. We consider an error a point p_e with either $p_e.value = KO$ and $p.key > p^*.key$ or $p_e.value = OK$ and $p.key \leq p^*.key$.

Therefore, the discriminant point is chosen between those that minimize the following function:

$$L_{OK}(x) = \{p | p.value = OK \wedge p.key \leq x.key\} \tag{3}$$

$$R_{KO}(x) = \{p | p.value = KO \wedge p.key > x.key\} \tag{4}$$

$$f(x) = |L_{OK}(x)| + |R_{KO}(x)| \tag{5}$$

From all the points that minimize the function $f(x)$ the ones with the smaller absolute difference between $|L_{OK}(x)|$ and $|R_{KO}(x)|$ are selected, so that the

Table 1. Algorithm cost compared with state-of-the-art solutions

	LCBM	**SGD** [6]	**ALS** [5]
time to model	$O(X)$	$O(X \cdot K \cdot E)$	$\Omega(K^3(N+M) + K^2 \cdot X)$
time to prediction	$O(1)$	$O(K)$	$O(K)$
memory usage	$O(N+M)$	$O(K(N+M))$	$O(M^2 + X)$

errors are balanced between OKs and KOs. The user QT value is the smallest key in this set. The solution can be found in polynomial time. The simplest approach is to pass three times over the points: one to compute $|L_{OK}(x)|$ for each point; one to compute $|R_{KO}(x)|$ for each point; one to find the point that minimizes $f(x)$.

Figure 3 shows an example where an user expressed 13 votes, 7 OK (blue circles) and 6 KO (red diamonds). The user QT value is 0.4, key of the discriminant point p^*. In fact, no other choice will deliver less than 4 errors (perfectly balanced) in the classification task (two with smaller keys and OK values and two with bigger keys and KO values).

The *worst case estimation* approach prevents inaccurate item profiles from corrupting the classification task. Indeed, without this mechanism KO votes with over-estimate item $MEAN$ would lead to over-strict QTs (OK votes with under-estimate item $MEAN$ values would lead to over-permissive QTs respectively).

4.4 Working Phase

In order to produce a prediction the algorithm simply compares QT values from the user profiler with $MEAN$ values from the item's rating PDF inference block. In particular, for each couple (u, i) without a rating it checks if the item $MEAN$ value is bigger than the user QT value. If this is the case the predicted user's rating for the item is OK, KO otherwise.

4.5 Algorithm Analysis

Table 1 reports the cost of the LCBM algorithm compared with costs from other state-of-the-art solutions. In the table X is the number of collected ratings, K is the number of hidden features [23] and E is the number of iterations. We remark that $O(\cdot)$ is an upper bound, while $\Omega(\cdot)$ is a lower bound for the computational complexity.

If we consider the time needed to calculate the model, our solution performs two passes over the set of available ratings, one for each functional block in the training phase, thus its linear asymptotical cost. Note that this is the lowest asymptotical cost possible to build the model (as any solution should read each available rating at least once to build a model). Once the model is built it will be constituted by a value for each item (its $MEAN$) and a value for each user (its QT), thus the occupied memory will be $O(N + M)$. Finally, calculating the prediction for a single couple (u, i) requires a single comparison operation over two values, and thus incurs a constant cost.

5 Experimental Evaluation

In this section we report the results of the experimental evaluation we conducted on a prototype implementation of our solution. The goal of this evaluation was to assess how much our solution is effective in predicting ratings and the cost it incurs in doing so.

5.1 Experimental Setting and Test Datasets

We implemented[1] our LCBM algorithm, and evaluated it against open-source implementations of batch based CF algorithms provided by the *Apache Mahout* project (mahout.apache.org). We compared LCBM against both *memory-based* and *matrix factorization* solutions, however, this section only reports results from the latter as memory-based solutions have well-known scalability issues [9], and our LCBM algorithm outperformed them both in prediction accuracy and computational cost[2]. We limited our comparative evaluation to *matrix factorization* solutions, focusing on the two factorization techniques presented in Section 2: SGD and ALS. More precisely, we considered a lock-free and parallel implementation of the SGD factorizer based on [6] (the source code can be found in the *ParallelSGDFactorizer* class of the *Apache Mahout* library); the algorithm makes use of user and item biases for the prediction task. These two values indicate how much the ratings deviate from the average. This intuitively captures both users tendencies to give higher or lower ratings than others and items tendencies to receive higher or lower ratings than others. We also considered a parallel implementation of ALS with *Weighted-λ-Regularization* based on [5] (the source code can be found in the *ALSWRFactorizer* class of the *Apache Mahout* library).

If not differently specified, we set the following parameters for the above algorithms: *regularization factor* $\lambda = 0.065$ (as suggested in [5]), $K = 32$ hidden features and $E = 30$ iterations. We defined this last parameter by noting that 30 was the lowest number of iterations needed for the prediction accuracy score to converge on the considered datasets. Note that *Apache Mahout* allows you to define additional optional parameters for the two algorithms. In our experiments we used the default values for these variables, embedded in the corresponding source code. The algorithms return a real value (between -1 and 1) as a preference estimation for a couple (u, i). To discretize the prediction we adopted the most natural strategy: if the result is positive or zero the algorithm predicts an OK, if negative a KO.

We used three test datasets for our comparative study. The first two datasets were made available by the *GroupLens research lab* (grouplens.org) and consist of movie rating data collected through the *MovieLens* recommendation website (movielens.org). The third one is the *Netflix prize* dataset

[1] Our prototype is available at `http://www.dis.uniroma1.it/~midlab/LCBM/`

[2] We also ran comparative tests with slope one algorithms. The results are not reported here as those algorithms showed worse performance than LCBM on all metrics.

(`http://www.netflixprize.com`). All the ratings in these datasets were on a scale from 1 to 5, and we "binarized" them as follows [24]: if the rating for an item, by a user, is larger than the average rating by that user (average computed over his entire set of ratings) we assigned it a binary rating of 1 (OK), -1 (KO) otherwise.

The experiments were conducted on an *Intel Core i*7 $2,4GHz$ quad-core machine with $20GB$ of memory, using a GNU/Linux 64-bit operating system.

5.2 Evaluation Methodology and Performance Metrics

Similar to most machine learning evaluation methodologies, we adopted a *k-fold cross-validation* approach. This technique divides the dataset in several folds and then uses in turn one of the folds as *test set* and the remaining ones as *training set*. The training set is used to build the model. The model is used to predict ratings that are then compared with those from the test set to compute the algorithm accuracy score. We randomly split the datasets in 5 folds, so that each fold contained 20% of the ratings for each item. The reported results are the average of 5 independent runs, one for each possible fold chosen as test set.

In general, in order to evaluate the results of a binary CF algorithm we can identify four possible cases: either (i) *correct predictions*, both for OKs (TP true positives), and KOs (TN true negatives) or (ii) *wrong predictions*, both if OK is predicted for an observed KO (FP false positives) or if KO is predicted for an observed OK (FN false negatives). These four values constitute the so called *confusion matrix* of the classifier.

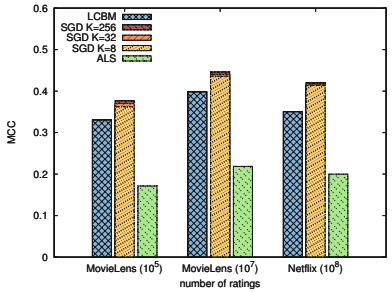
The *Matthews correlation coefficient* (**MCC**)[25] measures the quality of binary classifications. It returns a value between -1 and $+1$ where $+1$ represents a perfect prediction, 0 no better than random prediction and -1 indicates total disagreement between prediction and observation. The MCC can be calculated on the basis of the confusion matrix with the following formula:

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP) \times (TP + FN) \times (TN + FP) \times (TN + FN)}} \tag{6}$$

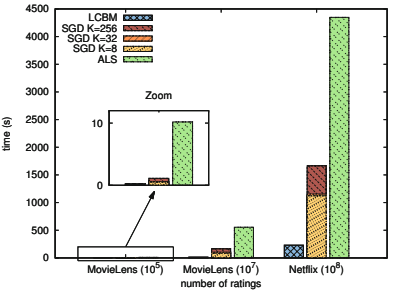
To assess the load incurred by the system to run the algorithms we also calculated the *time* needed to run the test (from the starting point until all the possible predictions have been made) and the peak *memory* load during the test. It is important to remark that running times depend strongly on the specific implementation and platform, so they must be considered as relative indicators, whose final scope is to reflect the asymptotic costs already presented in Table 1.

5.3 Evaluation Results

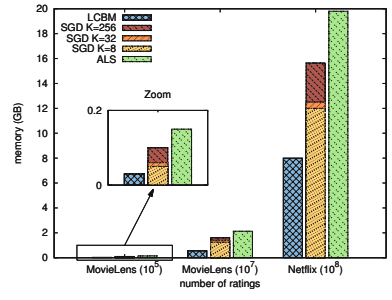
Figure 4 summarizes the performance of the CF algorithms over the three considered datasets, in terms of achieved prediction accuracy, time required for the prediction and memory occupation.



(a) MCC

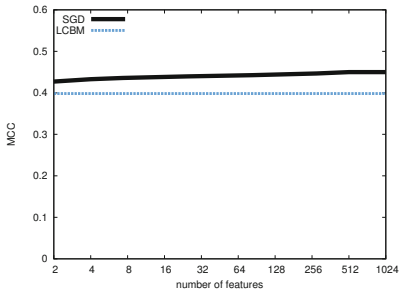


(b) Time

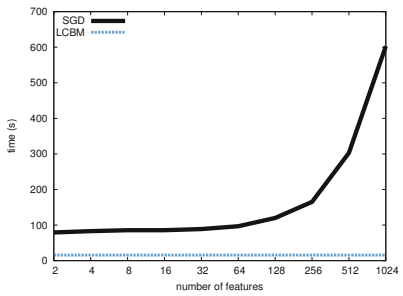


(c) Memory

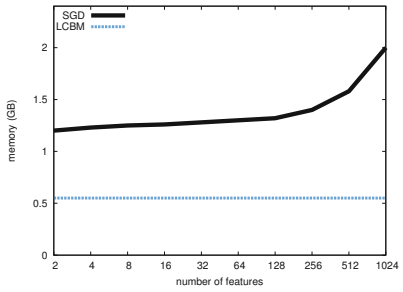
Fig. 4. Collaborative filtering algorithms performance, in terms of achieved accuracy, computational time required and memory occupation. The number of iterations for the matrix factorization models is set to 30. The SGD algorithm is trained with 8, 32 and 256 features. The number of features for the ALS algorithm is set to 32.



(a) MCC



(b) Time



(c) Memory

Fig. 5. LCBM vs. SGD performance varying the number K of hidden features, in terms of achieved accuracy, computational time required and memory occupation. The dataset used for the experiments is MovieLens with 10^7 ratings. The number of iterations was set to 30. The LCBM algorithm is agnostic to the number of features.

From Figure 4a it is possible to observe that LCBM consistently outperforms ALS by a large margin for all the considered datasets. Conversely, SGD outperforms LCBM in all datasets by a small margin whatever the value chosen for the number of features K is. By looking at this graph we can consider LCBM as a solution whose accuracy is very close to the accuracy offered by the best solution available in the state-of-the-art. However, the real advantages of LCBM come to light by looking at the load it imposes on the system.

Figure 4b shows the time required to conclude both the training and the test phases. Tests run with LCBM terminate much earlier that those run with SGD and ALS. This was an expected result as the time complexity of SGD is equivalent to the LCBM one only if we consider a single feature ($K = 1$) and a single iteration ($E = 1$) (cfr. Section 4.5). Note, however, that with this peculiar configuration SGD running time is still slightly larger than LCBM while its prediction accuracy, in terms of MCC, drops below the LCBM one (not shown in the graphs). The running time of ALS, as reported in the Figure, is always larger than LCBM.

The peak memory occupation is reported in Figure 4c. Also in this plot the gap between LCBM and MF techniques is evident. To summarize, LCBM is competitive with existing state-of-the-art MF solutions in terms of accuracy, but it runs faster while using less resources (in terms of memory).

The previous experiments have shown that the most performant matrix factorization solution is SGD. ALS, in facts, always showed the worst performance in our tests for all the considered metrics. Figure 5 reports the results of an experiment conducted on the MovieLens (10^7) dataset varying the number of hidden features K for the SGD factorizer. The LCBM performance are reported for comparison, and the corresponding curves are always constant because our solution is agnostic to K. Figures 5b and 5c show graphically what the asymptotic analysis has already revealed: time and space grow linearly with the number of features (note that the X-axis in the graphs has a logarithmic scale). The higher timeliness and memory usage thriftiness of LCBM is highlighted by the considerable gap between its curves and the SGD ones. Figure 5a reports the MCC values. As shown before SGD provides slightly better results than LCBM, and the gap tends to widen as the number of features grows. This, however, comes at the cost of a longer and more space consuming training procedure.

6 Conclusions

This paper introduced LCBM, a novel algorithm for collaborative filtering with binary ratings. LCBM works by analyzing collected ratings to (i) infer a probability density function of the relative frequency of positive votes that the item will receive and (ii) to compute a personalized quality threshold for each user. These two pieces of information are then used to predict missing ratings. Thanks to its internal modular nature LCBM is inherently parallelizable and can thus be adopted in demanding scenarios where large datasets must be analyzed. The paper presented a comparative analysis and experimental evaluation among LCBM

and current solutions in the state-of-the-art that shows how LCBM is able to provide rating predictions whose accuracy is close to that offered by the best available solutions, but in a shorter time and using less resources (memory).

References

1. Mangalindan, J.: Amazon's recommendation secret. CNN Money (2012), http://tech.fortune.cnn.com/2012/07/30/amazon-5/
2. Krikorian, R.: New tweets per second record, and how! Twitter blog (2013), https://blog.twitter.com/2013/new-tweets-per-second-record-and-how
3. Halevy, A., Norvig, P., Pereira, F.: The unreasonable effectiveness of data. IEEE Intelligent Systems 24(2), 8–12 (2009)
4. Narang, A., Gupta, R., Joshi, A., Garg, V.: Highly scalable parallel collaborative filtering algorithm. In: 2010 International Conference on High Performance Computing (HiPC), pp. 1–10 (2010)
5. Zhou, Y., Wilkinson, D., Schreiber, R., Pan, R.: Large-scale parallel collaborative filtering for the netflix prize. In: Fleischer, R., Xu, J. (eds.) AAIM 2008. LNCS, vol. 5034, pp. 337–348. Springer, Heidelberg (2008)
6. Takács, G., Pilászy, I., Németh, B., Tikk, D.: Scalable collaborative filtering approaches for large recommender systems. The Journal of Machine Learning Research 10, 623–656 (2009)
7. Gemulla, R., Nijkamp, E., Haas, P.J., Sismanis, Y.: Large-scale matrix factorization with distributed stochastic gradient descent. In: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (2011)
8. Zhuang, Y., Chin, W.S., Juan, Y.C., Lin, C.J.: A fast parallel sgd for matrix factorization in shared memory systems. In: Proceedings of the 7th ACM Conference on Recommender Systems, pp. 249–256. ACM (2013)
9. Su, X., Khoshgoftaar, T.M.: A survey of collaborative filtering techniques. Advances in Artificial Intelligence 4 (2009)
10. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Transactions on Knowledge and Data Engineering 17(6), 734–749 (2005)
11. Breese, J.S., Heckerman, D., Kadie, C.: Empirical analysis of predictive algorithms for collaborative filtering. In: Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence, pp. 43–52. Morgan Kaufmann Publishers Inc. (1998)
12. Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., Riedl, J.: Grouplens: an open architecture for collaborative filtering of netnews. In: Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work, pp. 175–186. ACM (1994)
13. Linden, G., Smith, B., York, J.: Amazon. com recommendations: Item-to-item collaborative filtering. IEEE Internet Computing 7(1), 76–80 (2003)
14. Lemire, D., Maclachlan, A.: Slope one predictors for online rating-based collaborative filtering. Society for Industrial Mathematics 5, 471–480 (2005)
15. Hu, Y., Koren, Y., Volinsky, C.: Collaborative filtering for implicit feedback datasets. In: Eighth IEEE International Conference on Data Mining, ICDM 2008, pp. 263–272. IEEE (2008)
16. Koren, Y.: Factorization meets the neighborhood: a multifaceted collaborative filtering model. In: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 426–434. ACM (2008)

17. Teflioudi, C., Makari, F., Gemulla, R.: Distributed matrix completion. In: ICDM, pp. 655–664 (2012)
18. Pan, R., Zhou, Y., Cao, B., Liu, N., Lukose, R., Scholz, M., Yang, Q.: One-class collaborative filtering. In: Eighth IEEE International Conference on Data Mining, ICDM 2008, pp. 502–511 (2008)
19. Ungar, L., Foster, D.P.: A formal statistical approach to collaborative filtering. In: CONALD 1998 (1998)
20. Wang, J., Robertson, S., de Vries, A.P., Reinders, M.J.: Probabilistic relevance ranking for collaborative filtering. Information Retrieval 11(6), 477–497 (2008)
21. Jsang, A., Ismail, R.: The beta reputation system. In: Proceedings of the 15th Bled Electronic Commerce Conference, pp. 41–55 (2002)
22. Huber, P.J.: The behavior of maximum likelihood estimates under nonstandard conditions. In: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, vol. 1, pp. 221–233 (1967)
23. Koren, Y., Bell, R., Volinsky, C.: Matrix factorization techniques for recommender systems. Computer 42(8), 30–37 (2009)
24. Das, A.S., Datar, M., Garg, A., Rajaram, S.: Google news personalization: scalable online collaborative filtering. In: Proceedings of the 16th International Conference on World Wide Web, pp. 271–280. ACM (2007)
25. Matthews, B.W.: Comparison of the predicted and observed secondary structure of t4 phage lysozyme. Biochimica et Biophysica Acta (BBA)-Protein Structure 405(2), 442–451 (1975)

Metamodel of a Logistics Service Map

Michael Glöckner, Christoph Augenstein, and André Ludwig

Information Systems Institute,
University of Leipzig
Grimmaische Str. 12, 04109 Leipzig, Germany
{gloeckner,augenstein,ludwig}@wifa.uni-leipzig.de
http://www.wifa.uni-leipzig.de/islog

Abstract. With the principle of division of labor in logistics, an integrator can focus on planning and monitoring within a network, while subsidiary logistics service providers (LSPs) are responsible for the actual physical manipulation of goods. Because of heterogeneous service descriptions, processes and IT-systems, the integrator requires a platform that provides the ability to interact with LSPs and to plan, execute and monitor contracts for integrator's customers. Such an integration platform is currently developed in the research project Logistics Service Engineering & Management. Crucial to such a platform is the ability to maintain a complete catalog and to efficiently identify and choose appropriate services. In this paper a metamodel-based approach is presented facing these requirements.

Keywords: Service Map, Metamodel, Logistics, Service Engineering and Management, Service Repository.

1 Introduction

Logistics is the applied science on executing orders by managing physical goods in a matter of space and time [1]. In a broader sense, it further deals with tasks of planning, operating and monitoring the systems that create physical goods and immaterial services. With big relevance of information exchange and automation in nowadays business also information flows grow more important in logistics. Accordingly, flows of both, physical goods and information, need to be considered in a comprehensive logistics system [1]. In consequence, new business models emerged in logistics industry. Most of these business models are based on a division of labor and of responsibility: logistics integrators (e.g. fourth party (4PL) or lead logistics service provider (LLP)) focus on planning and monitoring aspects of the flows of goods and information within a network of logistics providers. In contrast, process execution and actual physical manipulation of goods are realized by specialized logistics service providers (LSPs) acting as subcontractors for the logistics integrator [2, 3]. By combining offered services of the LSPs to composite services, the integrator is able to fulfill logistics contracts up to entire supply chains for its customers.

W. Abramowicz and A. Kokkinaki (Eds.): BIS 2014, LNBIP 176, pp. 185–196, 2014.

Confronted with low margins in logistics in general, an integrator has to choose the best available option for each task of a customer request. Thus, to plan and to operate a complex logistics service the integrator has to manage a variety of providers, their services and finally has to integrate with at least parts of their heterogeneous IT-systems of [1, 4, 5]. Each of the LSPs maintains its own systems, is capable of delivering a specific set of services and owns a specific set of resources in order to fulfill customer requests. Moreover, each LSP maintains a unique way of describing its services, thereby emphasizing different aspects of services and underlying concepts. To overcome this situation and to efficiently plan and operate a logistics contract, the integrator needs a solution to uniformly manage subsidiary providers as well as their systems and resources.

The Logistics Service Engineering & Management-platform (LSEM-platform) [6] makes use of the service-oriented design paradigm [7, 8] which helps to overcome some of the above aspects on a technological level. Modularization and loose-coupling of artifacts allow for a better exchangeability and fixed contracts allow for a more standardized way of describing interfaces in terms of necessary inputs and resulting outputs. As mentioned above, logistics is about handling goods and with this a service-oriented approach has to combine services from "the real world" and services which support the flow of goods by exchanging information between involved parties. In terms of service-oriented architectures (SOA) there are approaches addressing these difficulties when combining physical as well as non-physical services (for examples see: [9, 10]). However, on a more conceptual level in terms of describing the services themselves (e.g. handling of diverse service definitions or consideration of mutually exclusive service modeling approaches) further methods have to be developed.

Planning, operation and management of logistics contracts involve a multitude of providers and their services with differing service descriptions and resources. Thus, there is a need for a construction system which maintains a catalog of services and the originating providers and is moreover capable of supporting the integrator in order to efficiently identify and integrate adequate services for composite logistics services. A first draft towards a modular construction system for LSEM is already presented in [11] - the logistics service map (SM). It supports identification and integration of services on the LSEM-platform primarily at the beginning of a four-phase life-cycle. The logistics SM supports service composition in that it provides functionality for structuring, presenting and retrieval of services. Up to now, an appropriate metamodel for the integration of the SM-approach in the LSEM-platform is missing.

The contribution of this paper in particular is the development of such a metamodel for the logistics SM. The second section introduces the existing and to be developed parts of the LSEM-platform, that have essential influence on the metamodel. In section 3 related work is presented, compared to findings of section 2 and thus, provides further influence on the development described in section 4. After a critical appraisal, the paper ends with summary.

2 Logistics Service Engineering and Management

This section introduces parts of the LSEM-platform, a service life-cycle, the theoretical basics of a repository and its metamodel. Further, we focus on important characteristics of a service map concerning logistics issues. From these concepts we derive the integration constraints and essential criteria of the SM metamodel.

2.1 Service Life-Cycle

LSEM introduces a four-phase service life-cycle which supports a consistent and robust service development, allows for a sustainable execution and an orderly termination of logistics services [6]:

Servitization is the initial phase for developing atomic services. This phase includes aspects like 'analysis and design' [7], 'identifying and modeling' [12] or 'conceptualization and analysis' [13]. During this stage, the logistics integrator develops the basic services that are stored in the repository. Each LSP who wants to participate on the platform, registers itself and publishes the services he is capable of. Thus, this phase is not repeated on a regular base but only if new providers join the platform or if existing providers widen their service portfolio. The result of this phase is a set of atomic services the integrator uses to develop composite logistics services in order to fulfill customer contracts. The main issue here is to identify appropriate atomic services and their providers from a given portfolio of processes and capabilities.

Development involves all activities concerning the systematic composition of atomic services in order to fulfill customers' needs. Hence, facets like 'development and testing' [7], 'publishing' [12, 14], 'orchestration' [14] or 'development and testing' [13, 14] are regarded in this phase. In specific, the phase comprises modeling and simulation steps in order to construct and validate composite logistics services. The main concerns are to retrieve needed services (atomic or already composed) by an appropriate categorization as well as available providers, their associated resources and offered service level agreements (SLA).

Operation covers the field of implementation and actual execution. Correspondingly, aspects like 'deployment and execution' [12, 13], 'monitoring' [12, 14] or 'payment processing' [13] are contained in this field. During runtime, the integrator has to receive latest information about the current situation in its managed network. This information supports the operational management and error treatment. The main issues for this phase are the finance and accounting aspects as well as the monitoring aspects.

Retirement addresses the functions after the actual runtime of a service. This includes 'maintenance' [14] and 'retirement and rebinding' [13]. Further, a systematic performance analysis based on long-term monitoring for the evaluation of the LSP is done. This helps assessing the subcontractors on a long-term data base and provides an evident picture of their performance parameters for improved future planning issues.

Essential criteria, for the metamodel concept of the SM, are mainly focused in the first two phases that maintain a structured overview on the services of the platform, see also Fig. 1. We proceed with an overview of the influencing concepts and components of the LSEM-platform which support service engineering in particular and derive their impact on the metamodel.

2.2 Repository

The already mentioned service repository [15] of the platform is a crucial feature for managing necessary services and their descriptions. In the repository all artifacts related to services are stored and provided for platform tools in order to define, develop or monitor logistics services. On a *technical level*, the repository is a typical client-server solution. On the server side services and service models are stored in a content repository (Java Specification Request, JSR 170) which has a flat structure and no limitations regarding content to be added. The repository client is implemented using Java and components from the Eclipse framework. It allows browsing as well as synchronizing local working copies from platform tools. On a *conceptual level*, the repository and related components are part of the Service Modeling Framework (SMF) using a model-driven approach. The framework sees to make information about services in the repository available to the platform tools in that it interprets service models and extracts necessary information in order to create or update other service models. As a result, SMF provides a continuous modeling of services to platform users without the need of repeatedly modeling the same facts. Hence, we are able to provide support for LSEM-platform tools which are used in different phases of the life-cycle and which are used to add or update certain aspects of services like the process model, the interface definition or a textual description. These examples also show why a model-driven approach is necessary in order to uniformly handle these service aspects. They are heterogeneous in scope and in used language (modeling language). Thus, we need an approach which is capable of handling different types of service descriptions, i.e. models. For a more detailed explanation we refer to [15]. At its core, SMF is based on a metamodel, called common service model (CSM). The metamodel is kept simple and only consists of a few essential elements and their relationships, namely services, models, model elements and type information. With CSM we are able to interconnect models and model elements from different models, respectively. Purpose of the CSM is to uniformly interweave distinct service models each representing unique aspects of a service and thus on model-level allows for a generic and modular service model.

Development of a logistics SM is thus strongly related to SMF and its model-based concepts. Optimally, the SM also uses a model-driven approach so that information from early phases of the life-cylce can be transparently reused in later phases. Moreover, we emphasize the conceptual aspects of the repository as important for the metamodel development, see also Fig. 1. We now continue with the characteristics of the SM itself to identify further criteria for its metamodel.

2.3 Service Map for Service Engineering and Management

Offering a customizable approach for a logistics integrator, the logistics SM satisfies the needs for supporting the engineering and management of logistics services. It comprises functionality of both the addressed phases of the service life-cycle and the conceptual aspects of the repository, as shown in Fig. 1.

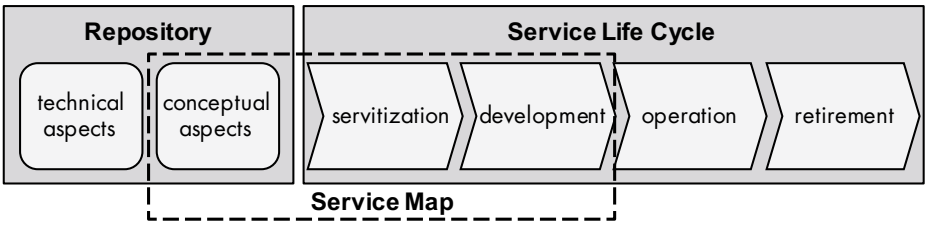


Fig. 1. Service Map addresses multiple phases and concepts in LSEM

The definition given in [11] outlines the emphasized phases by the functionality of a modular service construction system and the regarded relations between services. This implies the creation of atomic services (phase of servitization) that could be composed to composite services (phase of development). The conceptual aspects of the repository, like catalog function and the retrieval of services, are included with the structured categorization-pattern and the modular service construction functionality. Further, the SM includes different granularity levels and viewpoints from basic service description up to a category overview. Fig. 2 shows a simple example instance of a SM.

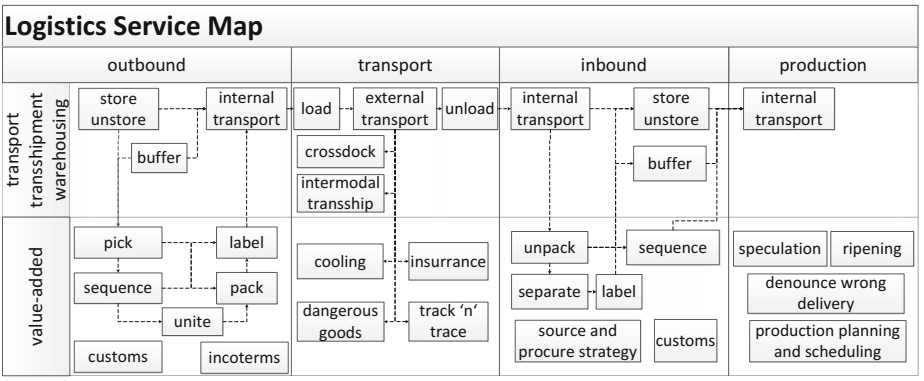


Fig. 2. Exemplary SM with two dimensions: 'classic logistics function vs. value-added' and 'stage-specific'. Dashed arrows mark compatible services for composition.

With this approach, a logistics integrator is supported in retrieving services in different use cases. (a) Adding a new LSP to the network and match its offered

services to the existing set of services in a logistics network by adding the new LSP to the provider list of the particular service. (b) Developing a new composite service to meet a specific customer's need by selecting and composing services from the SM. Service-specific information and attributes can be displayed when changing the selected granularity to a more detailed level to foster planning and monitoring. Moreover, the unique standard of the used set of services within a network and the visualization foster a precise mediation and communication between all stakeholders during the whole service life-cycle. (c) Finding compensational service or provider, when realizing the urgency for replanning or elimination of errors because of unpredictable disturbance in the network.

Consequently, a SM should be a core element of a service-oriented engineering and management platform and integrated by an appropriate metamodel due to the heterogeneous tools, models and platforms of the subcontractors.

2.4 Logistics

Since the integrator focuses on the engineering and management of logistics services in particular and a connection and composition of services in general only stands to reason within a distinct field of interest, a service map is always domain-specific. Blake [13] also proposes a domain-specific conceptualization and analysis in his presented service life-cycle. With the multitude of LSPs in the logistics industry [1, 4, 5] with their inherent multitude of provided services the catalog-function is emphasized once again. Another important aspect for a SM and the (potential) relations between the contained services are permission and refusal of particular service interrelations. The European Agreement concerning the International Carriage of Dangerous Goods by Road (ADR) [16] outlines a big quantity of self-explanatory examples for this fact.

3 Related Work

In the following section we discuss yet existing metamodel and SM approaches concerning their influence towards the parts of the LSEM-platform. After emphasizing the need for an appropriate metamodel, we outline approaches from current literature. The metamodeling section below provides examples of already present approaches which either have influence on or which are close to concepts of our approach. The service map section discusses the complex situation of the topic, found during literature studies.

3.1 Metamodeling

Atkinson and Kühne emphasize important requirements of model-driven development in [17] and thus, outline the capabilities of metamodeling. The most important capabilities in our context are the following, as they take on great significance especially in the context of logistics. Metamodeling approaches increase

the long-term productivity of primary software artifacts by reducing their sensitivity to changes. Those changes (and the resulting benefits of metamodeling in parenthesis) could be located in the fields of personnel (ease of understanding by different stakeholders) and functional requirements (integrating new features and capabilities with low maintainance and without disruption) and in development and deployment platforms (decoupling artifacts from tools with the inherent interoperability). However, issues may arise in this context: Dealing with models and metamodels may lead to multiple versions which are maintained independently and in the worst case lead to inconsistencies. To avoid such problems specialized platforms, so called metamodel-platforms, help to increase productivity when dealing with metamodeling issues. They offer the ability to manage metamodels and accordant versions of conformant models and in that they also allow versioning. The approach of [18] presents a metamodeling platform based on a model hierarchy and is explicitly dealing with modeling methods and their essential components like a modeling language, its notation, syntax and semantics, which in turn are also relevant for designing and implementing a service repository. The approach of [19] deals with modeling enterprise architecture with a layered strategy and therefore develops multiple, layer-specific metamodels and integrates them into a common model.

As the logistics integrator cooperates with a large number of LSPs and customers (personnel aspects), with a changing range of offered services and customer demands (functional requirements) and a widespread range of IT-systems (platforms), a metamodel hence, is an important artifact for tools that are integrated on the LSEM-platform. Accordingly, the logistics SM is obliged to own an appropriate metamodel itself. In [15] a metamodel for the integration and transformation of differing models has already been presented, yet. Its characteristics are compulsory to all integrated metamodels and subsequently to the SM-metamodel.

3.2 Related Service Map Concepts

When dealing with the topic of service maps, three characteristics can be described. (a) The term 'service map' is used and also the perception of functionality contains points of contact to our understanding of a SM, e.g. [20, 21, 22, 23]. (b) The term 'service map' is used, but a different contextual understanding is given, e.g. [24]. (c) The term 'service map' is not used explicitly, but the described concept contains notions similar to our context, e.g. [25].

Approaches of (a) are located in various fields. [20] provides the understanding closest to our context. The SM is used in the financial industry to get an overview of current portfolios to support merging and outsourcing of business models and IT-systems. Service retrieval and the creation of atomic or tailored customer-focused composite services is not an issue. [21] use a user-centric SM to visualize mobile apps within a 'user needs' categorization in order to identify 'empty' spaces with unsatisfied needs as potential service innovation opportunities. [22] propose a XML-based notion to enhance service structuring by establishing association and combination operators via XML-tags. [23] introduce a mobile data

management approach. With obtaining a detailed view of available networks and their inherent capabilities, attributes and offered services in the surrounding of a mobile device. However, their categorization pattern is strongly spatial-based, but also a comprehensive overview of available services is given from which the customer could choose its preference for specific purpose. The case (b) [24] addresses a mapping or matching, respectively, of Quality of Service (QoS)-classes. The approach deals with data quality in heterogeneous networks consisting of several network technologies. The goal is a mapping of performance parameters of the different technologies. The concept of (c), the 'service portfolio management framework' [25] combines both service science and portfolio management. Therefore, its purpose tends to a strategic understanding of service management rather than providing a modular construction system to integrate a number of subcontractors.

4 The Service Map Metamodel

The analysis of the parts of the LSEM-platform and the related work revealed the need for developing a metamodel for the logistics SM that considers the criteria outlined in section 2 and 3. This section now focuses the development of a metamodel for the logistics SM.

4.1 Conceptual Design

The SM supports the categorizing and development of services. Instances of the SM can be derived by the integrator from the metamodel to describe specific service portfolios of a network. The advantage of a metamodeling approach is a high abstraction that provides a high reusability in a wide range of cases and a simple interaction between several instances. To ensure compatibility to our research framework, the SM metamodel follows the same restrictions of SMF like all other models (i.e. based on the EMOF (Essential Meta Object Facility) compatible Ecore[1] metametamodel of the Eclipse Foundation). Having all models defined with the same modeling language on metamodel-level, we are able to reuse information contained in these models. Thus, the SM metamodel is also defined in Ecore, but could be easily implemented in other frameworks as well. The metamodel does not raise claim to completeness and is adaptable. The following metamodel is situated on the M^2-level, whereas M^0 is the original SM (i.e. service catalog and construction system) and M^1 designates a model of the SM (e.g. Fig. 2).

Fig. 3 depicts the current version of the SM metamodel. Each instance consists of exactly one *catalog* containing services available to the integrator on the platform. This catalog is structured using *categories* which depend on a specific *domain* (e.g. logistics). Thus, the catalog represents a structured list of *services*, each capable of one or more *capabilities*. These capabilities belong to a specific

[1] `http://www.eclipse.org/modeling/emf/?project=emf`

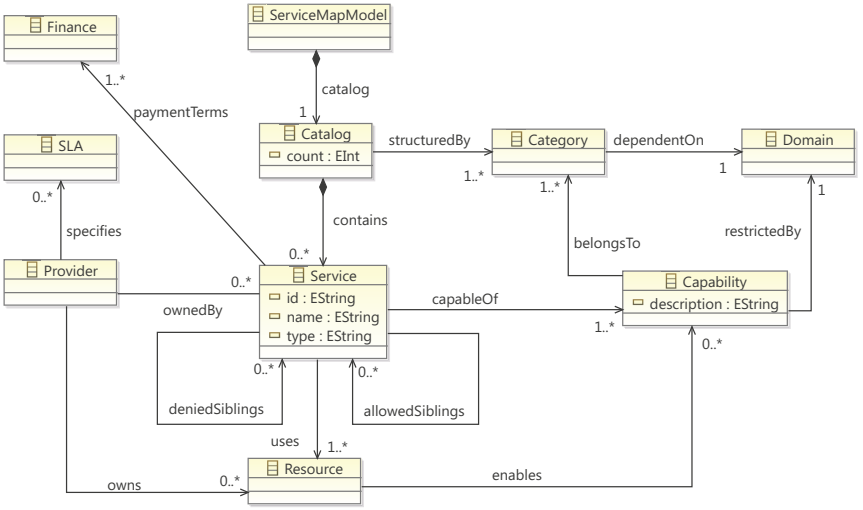


Fig. 3. Ecore model version of SM metamodel

category and are restricted by the concrete domain. For instance, on a high level capabilities represent the ability to transport, store or to fulfill more complex composite and value adding services. In order to provide capabilities in terms of services, a *provider* owns specific *resources* like trucks or warehouses which are consumed during service execution but typically are available again afterwards. Each provider is also allowed to specify zero or more *service level agreements (SLA)* for its services in which it specifies constraints of service provisioning and terms of payment. Finally, services can either depend on other services or are restricted not to work with other services. Therefore, each service contains references to others which are either available for the definition of a composed service (*allowedSiblings*)or not (*deniedSiblings*).

An instance of a logistics SM thus represents a complete list of capabilities (represented by services) of the provider network, including services the integrator can provide on its own. Hence, the service map serves as a catalog of available services. Moreover, during the creation of a complete logistics service for a customer, the service map also serves as a unique point of information and as a reference for searching appropriate services and providers. This becomes apparent in the development phase in particular. During rough planning of a logistics service, the service chain has to be constructed by choosing suitable services. According to customers' requirements, appropriate providers have to be chosen for each task in the service chain. Therefore, the service map is used to identify providers who offer the needed service type.

Because the logistics SM follows a metamodel-based approach, an integrator also has the ability to manage multiple provider networks independently, for instance in automotive industry. Requirements of OEMs (Original Equipment

Manufacturer) are very strict in that they often demand closed supply chains. Providers are not allowed to share their resources between different contracts. For instance, an integrator responsible for warehouses with vendor managed inventory (VMI) for multiple OEMs at nearby production sites is liable to provide warehouse resources to each of the OEM exclusively, i.e. separate infrastructure and employees. With this in mind, an integrator is still able to optimally allocate resources if he partitions its complete network into independent parts and manages each of them separately. Though, same services are in different catalogs, the integrator is aware of the total resources available and can create an efficient supply chain for each customer.

4.2 Discussion

The integration of interfaces as an aspect of the SM metamodel was roughly discussed during design process. However, it forms a relevant notion, but we decided to leave the topic out in the current version. Due to the CSM-functionality of the repository (see 2.2), interweaving with other models and tools is granted. Further a capability-centered approach was considered. When building composite services and supply chains, the inherent service function or capability is the important object for the integrator or planer, respectively, as these functions realize the actual flow of goods and information. On the contrary, the SOA design-paradigm always focuses on the services themselves. Hence, with the service class as the obligatory central component of every model and metamodel, respectively, the service is put in the focus of attention. Consequently a unique connection point is ensured in every case and every part of the architecture and the related model-driven approaches. However, the developed metamodel derives its structure and content from the example-domain of logistics, but excludes logistics-specific aspects by incorporating them in an abstract way. Through including a certain domain as a crucial foundation of a SM, the presented approach is also usable in other fields of service-oriented industries.

5 Summary

In this paper we presented an approach for metamodel-based service map to be used in logistics. In contrast to [11] important concepts and used technologies of the logistics SM have been developed and are more elaborate. The approach is designed to the needs of the LSEM-platform and is compatible to other tools and concepts. Most important, the logistics SM is able to fill the gap of categorizing, structuring and identifying available services on the platform and hence is essential in the early phases of the service life-cycle. We initially presented constraints and related tools of the platform like the service repository and proceeded with basic principles a service map is developed for. We also localized this approach in the logistics domain and could thus tailor the service map to the specific needs of a logistics integrator. Nevertheless, the approach is also applicable in other service-oriented industries.

Acknowledgement. The work presented in this paper was funded by the German Federal Ministry of Education and Research under the project LSEM (BMBF 03IPT504X).

References

[1] Gudehus, T., Kotzab, H. (eds.): Comprehensive Logistics. Springer, Heidelberg (2012)
[2] Schmitt, A.: 4PL-Providing^{TM}als strategische Option für Kontraktlogistikdienstleister: Eine konzeptionell-empirische Betrachtung. Deutscher Universitäts-Verlag / GWV Fachverlage GmbH, Wiesbaden (2006)
[3] Kutlu, S.: Fourth party logistics: The future of supply chain outsourcing? Best Global Publishing, Brentwood (2007)
[4] Handfield, R., Straube, F., Pfohl, H.C., Wieland, A.: Trends and Strategies in Logistics and Supply Chain Management: Embracing global logistics complexity to drive market advantage. Trends and strategies in logistics and supply chain management. DVV Media Group, Hamburg (2013)
[5] Terry, L.: 2014 third-party logistics study: The state of logistics outsourcing: Results and findings of the 18th annual study (2014)
[6] Klinkmüller, C., Kunkel, R., Ludwig, A., Franczyk, B.: The logistics service engineering and management platform: Features, architecture and implementation. In: Abramowicz, W. (ed.) BIS 2011. LNBIP, vol. 87, pp. 242–253. Springer, Heidelberg (2011)
[7] Erl, T.: SOA: Principles of service design. Prentice Hall, Upper Saddle River (2008)
[8] Papazoglou, M.: Web services: Principles and technology. Pearson/Prentice Hall, Harlow (2008)
[9] Beverungen, D., Knackstedt, R., Müller, O.: Entwicklung serviceorientierter architekturen zur integration von produktion und dienstleistung – eine konzeptionsmethode und ihre anwendung am beispiel des recyclings elektronischer geräte. Wirtschaftsinformatik 50(3), 220–234 (2008)
[10] Acharya, M., et al.: Soa in the real world – experiences. In: Benatallah, B., Casati, F., Traverso, P. (eds.) ICSOC 2005. LNCS, vol. 3826, pp. 437–449. Springer, Heidelberg (2005)
[11] Glöckner, M., Ludwig, A.: Towards a logistics service map: Support for logistics service engineering and management. In: Blecker, T. (ed.) Pioneering Solutions in Supply Chain Performance Management: Proceedings of the Hamburg International Conference of Logistics (HICL) 2013, Eul, Lohmar, Köln. Reihe: Supply chain, logistics and operations management, vol. 17, pp. 273–285 (2013)
[12] Papazoglou, M.P., Van Den Heuvel, W.-J.: Service-oriented design and development methodology. International Journal of Web Engineering and Technology 2(4), 412 (2006)
[13] Blake, M.B.: Decomposing composition: Service-oriented software engineers. IEEE Software 24(6), 68–77 (2007)
[14] Gu, Q., Lago, P.: A stakeholder-driven service life cycle model for soa. In: Crnkovic, I. (ed.) 2nd International Workshop on Service Oriented Software Engineering: in Conjunction with the 6th ESEC/FSE Joint Meeting, pp. 1–7 (2007)
[15] Augenstein, C., Ludwig, A., Franczyk, B.: Integration of service models - preliminary results for consistent logistics service management. In: 2012 Annual SRII Global Conference, pp. 100–109. IEEE Computer Society, Los Alamitos and Calif (2012)

[16] United Nations: ADR - European Agreement Concerning the International Carriage of Dangerous Goods by Road: Applicable as from 1 January 2013. United Nations, New York (etc.) (2012)
[17] Atkinson, C., Kühne, T.: Model-driven development: a metamodeling foundation. IEEE Software 20(5), 36–41 (2003)
[18] Karagiannis, D., Kühn, H.: Metamodelling Platforms, vol. 2455, pp. 451–464. Springer, Heidelberg (2002)
[19] Braun, C., Winter, R.: A comprehensive enterprise architecture metamodel and its implementation using a metamodeling platform. In: Enterprise Modelling and Information Systems Architectures, vol. P-75, pp. 64–79. GI (2005)
[20] Kohlmann, F., Alt, R.: Aligning service maps - a methodological approach from the financial industry. In: Sprague, R.H. (ed.) Proceedings of the 42nd Annual Hawaii International Conference on System Sciences, pp. 1–10. IEEE Computer Society Press, Los Alamitos (2009)
[21] Kim, J., Lee, S., Park, Y.: User-centric service map for identifying new service opportunities from potential needs: A case of app store applications. Creativity and Innovation Management 22(3), 241–264 (2013)
[22] Vaddi, S., Mohanty, H., Shyamasundar, R.: Service maps in xml. In: Potdar, V. (ed.) Proceedings of the CUBE International Information Technology Conference, pp. 635–640. ACM, [S.l.] (2012)
[23] Kutscher, D., Ott, J.: Service maps for heterogeneous network environments. In: MDM 2006, Japan. IEEE Computer Society, Los Alamitos (2006)
[24] Ryu, M.S., Park, H.-S., Shin, S.-C.: Qos class mapping over heterogeneous networks using application service map. In: Networking, International Conference on Systems and International Conference on Mobile Communications and Learning Technologies, vol. 13. ICN (2006)
[25] Kohlborn, T., Fielt, E., Korthaus, A., Rosemann, M.: Towards a service portfolio management framework. In: Australian Conference on Information Systems, ACIS2009, pp. 861–870 (2009)

Yet another SLA-Aware WSC System

Dmytro Pukhkaiev[1], Tetiana Kot[1], Larysa Globa[1], and Alexander Schill[2]

[1] National Technical University of Ukriane «Kyiv Polytechnic Institute», Information Telecommunication Networks Department, Peremoga Ave. 37, 03056 Kyiv, Ukraine
dpukhkaiev@stud.its.kpi.ua, {tkot,lgloba}@its.kpi.ua
[2] Dresden University of Technology, Faculty of Computer Science, Nöthnitzer Str. 46, 01187 Dresden, Germany
alexander.schill@tu-dresden.de

Abstract. Web Service Composition is a convenient way to save programming effort by reusing existing web services to create complex applications. A very important task in this context is to satisfy quality of a service (QoS) constraints. An easy and efficient way to store QoS requirements is to use service level agreements (SLA). Most state of the art methods for SLA-aware WSC focus only on some narrow aspect such as speed of composition or user feedback, but other aspects are often neglected. This paper presents SLA-aware WSC System which simultaneously solves most of WSC problems.

Keywords: web services, QoS, SLA, composition.

1 Introduction

Nowadays, the concept of SOA (Service Oriented Architecture) is widely acknowledged by scientific society. Vast amount of researches have been conducted although there are still enough questions to be investigated. Composite Web Services provide client with the ability to solve complex business tasks without much programming effort. Web Service Composition (WSC) presumes having at least one web service (often much more) for each simple task in corresponding business process. These services are equal on functional parameters but have to be evaluated in terms of best compliance for the overall application.

An efficient approach is to use QoS requirements in order to distinguish such web services. Such non-functional parameters as performance, reliability, accessibility, availability, scalability, cost etc. dramatically effect on user experience and should be taken into consideration on the design stage of application.

Service Level Agreement (SLA)-aware composition of web services allows solving this task. SLA is a convenient way to present and evaluate non-functional parameters both on stages of execution and monitoring. Represented by Web Service Agreement (WSA) it provides information about functional and non-functional parameters of the web service. However, contemporary SLA-aware WSC approaches have many issues. The most significant one is a lack of broadness. Focusing on a narrow task (e.g. speed of composition) other important aspects are neglected.

W. Abramowicz and A. Kokkinaki (Eds.): BIS 2014, LNBIP 176, pp. 197–205, 2014.

This results in having a method which solves a minor issue, but ignores an initial goal – providing the user with an ability to compose a service by reusing existing web services. SLA-Aware WSC System [1] allows performing WSC with respect to QoS requirements and provides dynamic reconfiguration of a running composite web service. Non-functional parameters violation is a common situation so it should be handled automatically. Otherwise, it will imply into a significant degradation or even a failure of the entire application.

The remainder of the paper is structured as follows. Section 2 contains a brief background needed to formulate the problem – an analysis of contemporary SLA-aware WSC approaches has shown that various approaches cannot handle a problem with having full stack of QoS parameters alongside with subjective QoS and monitoring support. Section 3 provides a brief description of SLA-aware WSC System with various extensions like floating coefficients and price/performance evaluation features added. It also covers such topic as an integral indicator of web service quality compliance. Section 4 summarizes the work performed and provides an outlook on future work.

2 Problem Definition

Composite web service which is the goal of WSC is a complex software application. SLA-Aware WSC which is the topic of this paper is only one of the steps in order to achieve this goal. In this section a brief overview of other steps (e.g. design, execution stages of Business Process Management (BPM)) in context of automatic application development is presented. Another topic of this section is SLA-Aware WSC algorithms analysis with their strengths and weaknesses such as subjective QoS parameters neglection, full stack of QoS parameters support and no monitoring mechanism lightened.

2.1 Background

Design stage of BPM is represented by graphical standards such as BPMN [2], USDL [3], UML [4]. These notations depict workflow in diagrammatical way.

Translating graphical diagrams into code which can be interpreted by machine is an important step in automatic applications development. Various methods and tools have been created to overcome this problem [5], [6].

Workflow enactment is a stage when workflow is represented by service orchestration [7]. This means having an algorithm of services invocations, their relations. WS-BPEL [8] is de-facto a main standard for describing a service on the workflow enactment stage.

WS-BPEL file can be abstract and executable. While abstract BPEL conceals specific information e.g. web services used in the orchestration, executable BPEL file can be easily interpreted by the BPEL engine and executed. Web services location and composition are meant to find corresponding web services and create a composite one respectively.

On stage of workflow analysis the composite service is constantly monitored to ensure stable and proper work of the application.

2.2 SLA-Aware Algorithms Analysis

Numerous researches have been recently conducted on the topic of WSC [1]. However the topic of SLA-aware or QoS-aware WSC in particular has less number of researches. In this subsection an evaluation of SLA-aware approaches is conducted. They are trying to optimize WSC from different perspectives.

In [9, Zhou et al] authors present preference-based approach which calculates an overall QoS of a composite web service regarding four QoS parameters such as price, response time, reliability and reputation. User preferences are used as coefficients for these parameters, if parameter has more value for the user it will result to its higher influence on overall QoS. This approach lacks support of other QoS parameters, monitoring phase.

In [10, Berbner et al] heuristic approach is presented. This approach divides QoS parameters into three groups: additive parameters, multiplicative parameters and attributes aggregated by Min-operator. These groups cover much more QoS parameters than preference-based approach although it doesn't support subjective QoS requirements that represent user feedback. SLA violations monitoring is provided by the presented architecture.

In [11, Mardukhi et al] authors present genetic algorithm to solve QoS-aware WSC problem. They use decomposition of global QoS constraints into local ones for every web service in the composition. After that linear search is used for simple web services in order to choose the best one. QoS parameters are divided into two main groups: positive and negative. Positive parameters such as availability and throughput should be maximized while negative like price and response time should be minimized. The main focus of this approach is on computational time of composition. It implies into reduction of the WSC quality because of the neglection of such features as subjective QoS or parameters extensibility. Although good performance during runtime and hence the possibility of SLA violations monitoring is stated no mechanism is presented.

In [12, Aiello et al] a system to perform WSC is presented. It uses Breadth First Use algorithm to perform the composition. Utilization of only such QoS parameters as response time and throughput significantly deteriorates overall quality of the composition. No monitoring phase is introduced as well.

Comparing these approaches it is evident that most of them do not cover a sufficient number of QoS subjective parameters. This condition is essential for achieving an optimal composite web service. Another important issue is lack of the monitoring phase in all approaches except [10] heuristic approach. This means that resulting service will fail and unsatisfy its customer in case of any component web service failure. At the same time the most reliable in this comparison heuristic approach lacks flexibility such as new user-defined QoS parameters addition, objective QoS parameters support and user preferences.

Table 1 summarizes the results of comparison.

Table 1. SLA-aware WSC approaches comparison

	Preference-based [9]	Heuristic [10]	Genetic [11]	Breadth First Use [12]
Full stack of QoS parameters	-	+	+	-
Subjective QoS	+	-	-	-
Monitoring	-	+	+/-	-

Thus, SLA-aware WSC approach which is able to stand up to all the requirements provided in this section is a topical task.

3 SLA-Aware WSC System

In [1] the concept of SLA-Aware WSC System was presented. This paper extends it in terms of user preferences e.g. floating coefficients for parameters, price/performance evaluation. Moreover, this version of SLA-Aware WSC System is more practical oriented than the previous one – in this paper an example of the common scenario for the system is given.

3.1 System Description

An SLA-Aware WSC System is a part of automatic web-oriented applications development tool. It presumes having an input file with services' descriptions represented in the proper way to find corresponding web services. System consists of five major blocks: Service Locator, SLA Extractor, Decision Maker, Service Combiner and Service Monitor.

Service Locator block is intended to find web services which meet functional parameters provided by input BPMN (BPEL) file. Web services which were found for each activity stated in BPMN are organized in the list sorted by relevance. In our context relevance means achievement of non-functional properties stated by the client.

Organizing web services into the list means their evaluation by non-functional properties. This evaluation is performed by cooperation of blocks named earlier. SLA Extractor block finds QoS information in WS-Agreements attached to web service and provides Decision Maker with this information. It calculates rankings based on rules found in ontology or user preferences provide by the client personally. User-defined QoS preferences have higher priority than ontology rules. This means that overall ranking will be calculated with respect to the clients demands thus fulfilling Subjective QoS parameters support constraint. Service Combiner creates an output executive BPEL file with all web services invoked and ready to execution.

Service Monitor constantly evaluates QoS parameters changes and reconfigures composite web service in case of their violations.

3.2 Integral Indicator of Web Service Quality Compliance

This subsection provides details for evaluating composite services. Overall QoS ranking or integral indicator of web service quality compliance is the key parameter in services' evaluation. It shows which service satisfies QoS preferences better. Comparing to the QoS of a single web service ranking of a composite one is not a trivial task. Composition of QoS parameters depends on a workflow described by client.

Table 2. QoS computations and rankings

QoS Parameter	R	WS Composition Patterns			
		Sequence	Parallel	Switch	Loop
Reliability	1	$\prod_{i=1}^{m} Rl_i$	$\prod_{i=1}^{m} Rl_i$	$\prod_{i=1}^{m} p_j Rl_i$	Rl_i^k
Performance	2	$\sum_{i=1}^{m} P_i$	$\max(P_i)$	$\sum_{i=1}^{m} p_j P_i$	P_i^k
Availability	3	$\prod_{i=1}^{m} Av_i$	$\prod_{i=1}^{m} Av_i$	$\prod_{i=1}^{m} p_j Av_i$	Av_i^k
Accessibility	4	$\prod_{i=1}^{m} Ac_i$	$\prod_{i=1}^{m} Ac_i$	$\prod_{i=1}^{m} p_j Ac_i$	Ac_i^k
Robustness	5	$\prod_{i=1}^{m} Rb_i$	$\prod_{i=1}^{m} Rb_i$	$\prod_{i=1}^{m} p_j Rb_i$	Rb_i^k
Accuracy	6	$\prod_{i=1}^{m} Acr_i$	$\prod_{i=1}^{m} Acr_i$	$\prod_{i=1}^{m} p_j Acr_i$	Acr_i^k
Scalability	7	$\max(Sc_i)$	$\sum_{i=1}^{m} Sc_i$	$\sum_{i=1}^{m} p_j Sc_i$	Sc_i
Capacity	7	$\max(Ca_i)$	$\sum_{i=1}^{m} Ca_i$	$\sum_{i=1}^{m} p_j Ca_i$	Ca_i

In Table 2 QoS parameters computations for a composite service are presented. Column R represents rank of the corresponding QoS requirement regarding its influence on the composite service. The rank of the corresponding QoS parameter influences on a value of the integral indicator of web service quality compliance which represents an overall measurement of composite web service's QoS. Thus, the same composite web service will have different overall position in the priority list if ranking of single QoS parameter changes e.g. specified by user. Default rankings are chosen corresponding to the need of general contemporary internet services. If e.g. service has more and more clients and scalability becomes more important factor it can be re-prioritized by user.

Applying floating parameters i.e. user-defined rankings of QoS parameter influence overrides these rankings in order to provide a service which satisfies user requirements in the best possible way. Whether client does not want to specify any properties default values of rankings will be applied. Comparing to [1] this extension allows utilizing user feedback, thus satisfying subjective QoS parameters.

Thus, formula for integral indicator of web service quality compliance:

$$Nf = \text{Operator}(R_i QoSp_i). \quad (1)$$

Where $QoSp_i$ - one of the QoS parameters listed in Table 2, R_i - ranking of corresponding QoS parameter. $QoSp_i$ has a value from 0 to 1 proportionally to the actual value of the parameter. Operator can be the sum, multiplication, max or power operator depending on the workflow.

3.3 Application Scenario for SLA-Aware WSC System

This subsection provides a detailed example of a possible scenario for using the presented approach and clarification of software tool based on this example.

Let us assume a person who uses the tool for dynamic WSC (provider) has a goal to develop and provide service which helps its consumers to book a fully customizable vacation with having a hotel, flight, taxi and cultural events pre-booked. He is not able to program such a service or there is no much time for the development. Thus, using existing web services which can partially provide necessary functionality is a convenient option for fast application development.

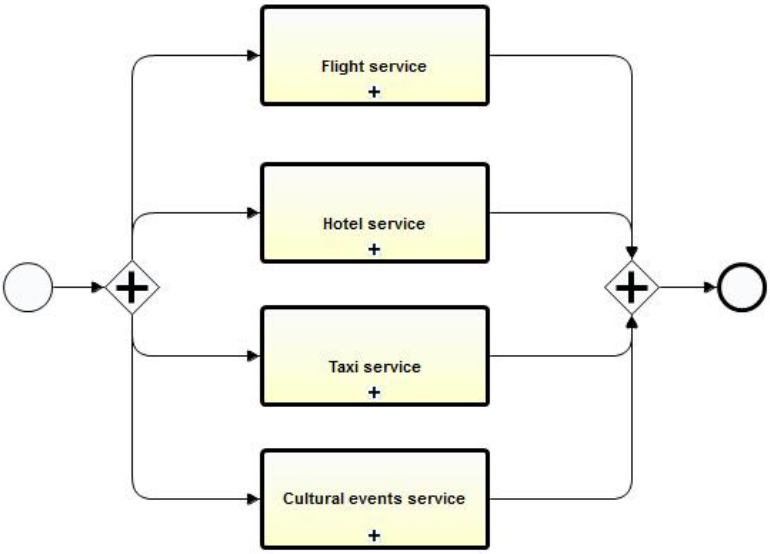


Fig. 1. Simplified BPMN diagram of provider's application

Provider has various constraints regarding his tool, e.g. response time, cost etc. After successful registration in WSC System, provider can either create a new Project or edit an existing one. After choosing an appropriate option, provider can upload BPMN or BPEL file into the system for analysis. At this stage QoS constraints should be specified. Otherwise Service Locator will search for any services, satisfying functional parameters, extracted from uploaded BPEL (BPMN) file. WS-Agreement for the composite service is generated if QoS restrictions have been specified by provider.

On Fig. 1 BPMN diagram for Vacation Service is presented. Concrete workflows are omitted for simplicity.

Service locator extracts the information about functional parameters from BPEL-file which was either uploaded or generated from BPMN. It also searches the appropriate services in UDDI or service brokers (considering functional parameters).

Now provider has a list with sets of web services which are capable to reach provider's goal (in the given example different combinations of services, providing flight, hotel, taxi and cultural events booking, are presented). They are sorted by the integral indicator of web service quality compliance by default. The sorting is done transparently by Decision Maker module.

A new feature comparing to [1] is that price/performance is evaluated. This means that if service with the best ranking has the price considerably higher than the second one in the list, price/performance will be calculated. Instead of picking the best service by ranking it will be chosen by price/performance.

Provider can set whether stage of picking the best services needs human interaction (i.e. choosing a web service) or it should be done automatically. The last case means that the best service regarding integral indicator of web service quality compliance would be applied as default. Then chosen services are purchased. And provider has a functioning composite web service.

The suggested WSC Tool contains built-in monitoring and dynamic reconfiguration module which provides fast and reliable service reconfiguration. This module utilizes previously composed list of possible services.

QoS parameters of composite web service are under constant observation by WSC System. Let us assume that one of these parameters, for instance response time, is not appropriate for some period of time. QoS parameters of all services within the composite are re-evaluated and inappropriate one is identified. After that provider receives a notification with options to reconfigure struggling web service i.e. choose a web service which satisfies non-functional parameters better than the previous one. Reconfiguration is performed on flight automatically if provider has chosen the option of automatic reconfiguration in system settings. In case of violating functional parameters, composite web service is recomposed from scratch – violating functional parameters cannot be allowed, this means that application is not running properly.

In the case when web service for booking a hotel has less availability than specified by provider, the overall composite service does not satisfy QoS constraints. WSC System checks availability parameters of all four individual web services and identifies that hotel service is struggling. WSC System still contains the list of potential services for provider's task. These services can be compared to a failed web

service in terms of QoS parameters. Manually or automatically unsatisfying hotel web service is replaced by the most appropriate one from the list. Then WSC System re-assembles composite web service. Provider again has a service which is fully functional and satisfies all requirements.

End user is provided with the web interface which gathers information from various web services thus being able to compose a customized vacation.

4 Conclusion and Future Work

This paper presents an SLA-aware WSC System which is able to overcome major problem of most WSC approaches – narrow task focusing and thus neglection of other important aspects. SLA-aware WSC System provides this by covering such aspects as full stack of QoS parameters support, subjective QoS i.e. user preferences, monitoring stage support.

Results of the system run cannot be presented in common way like time diagrams etc., because the goal of this system is in another aspect. It does not have to be better than other algorithms in terms of speed or memory consumption, but it plays an important role of an approach which can unite all QoS constraints in one system, thus being able to take part in automatic application development.

Future work will focus on writing a tutorial for SLA-aware WSC System and implementing it into automatic application development tool which is being created now. Another important aspect is to test the implementation and thus evaluate correctness of solution in terms of providing user with the best possible result. In case of negative outcome in any test refinements will be suggested and implemented.

References

1. Pukhkaiev, D., Kot, T., Globa, L., Schill, A.: A novel SLA-aware approach for web service composition. In: IEEE EUROCON, pp. 327–334 (2013)
2. OMG Business Process Model and Notation, `http://www.omg.org/spec/BPMN/2.0/PDF`
3. Oberle, D., Bhatti, N., Brockmans, S., Niemann, M., Janiesch, C.: Countering Service Information Challenges in the Internet of Services. Journal of Business & Information System Engineering 1, 370–390 (2009)
4. Russell, N., van der Aalst, W.M.P., ter Hofstede, A.H.M., Wohed, P.: On the suitability of UML 2.0 activity diagrams for business process modeling. In: Proceedings of the 3rd Asia-Pacific Conference on Conceptual Modeling, APCCM 2006, vol. 53, pp. 95–104. Australian Computer Society, Inc., Darlinghurst (2006)
5. Puhkaev, D., Kot, T.: Conversion of business-processes of extended BPMN into BPEL code. In: CriMiCo 2012, pp. 411–412 (2012)
6. Ouyang, C., van der Aalst, W.M.P., Dumas, M., ter Hofstede, A.H.M., Mendling, J.: From Business Process Models to Process-Oriented Software Systems. ACM Transactions on Software Engineering and Methodology, 1–37 (2009)
7. Foster, H., Uchitel, S., Magee, J., Kramer, J., Hu, M.: Using a rigorous approach for engineering Web service compositions: a case study. In: Proceedings of the 2005 IEEE International Conference on Services Computing (SCC 2005), pp. 217–224. IEEE Computer Society, Washington, DC (2005)

8. OASIS Web Services Business Process Execution Language, http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.pdf
9. Zhou, W., Wena, J., Gaob, M., Liu, J.: A QoS preference-based algorithm for service composition in service-oriented network. Optik - International Journal for Light and Electron Optics 124(20), 4439–4444 (2013)
10. Berbner, R., Spahn, M., Repp, N., Heckmann, O., Steinmetz, R.: Heuristics for QoS-aware Web Service Composition. In: Proceedings of the IEEE International Conference on Web Services 2006, pp. 72–82. IEEE Computer Society, Washington, DC (2006)
11. Mardukhi, F., NematBakhsh, N., Zamanifar, K., Barati, A.: QoS decomposition for service composition using genetic algorithm. Applied Soft Computing 13(7), 3409–3421 (2013)
12. Aiello, M., el Khoury, E., Lazovik, A., Ratelband, P.: Optimal QoS-Aware Web Service Composition. In: IEEE Conference on Commerce and Enterprise Computing, pp. 491–494 (2009)

Processes within the Context of Cloud Operations Management
A Literature Reflection

Christian Schulz and Klaus Turowski

Otto von Guericke University Magdeburg
Very Large Business Applications Lab.
P.O. Box 4120
39016 Magdeburg, Germany
{christian.schulz,klaus.turowski}@ovgu.de
http://mrcc.eu

Abstract. Virtualization is a key technology for current data centers to consolidate their resources in order to lower the total cost of ownership benefiting from the economies of scale. Cloud computing becomes an increasingly established technology enabling providers to realize that vision. Meeting the technical requirements of Clouds while staying flexible against changing business requirements new challenges to operations management in Cloud data centers are emerging. Standard interfaces and holistic approaches to manage operational processes and infrastructure resources are still missing. Hence, this paper reflects the current state of the art in the context of Cloud operations management to obtain an overview of existing approaches and applied operation processes on the one hand and to examine the need for further research on the other hand. It is demonstrated that such a research gap does exist since the existing literature is not considering the whole context of Cloud operations management yet.

Keywords: Cloud Operations Management, Virtual Resource Management, Virtualized Data Center.

1 Introduction

Nowadays, computing comes to be a separate, fifth public utility besides water, electricity, gas and telephony [4] as it has been already prophesied by John McCarthy in 1969. One good bet for realizing that vision is the Cloud computing paradigm. Cloud computing supports a so-called elasticity which signifies a dynamic scalability of pooled physical resources to deploy apparently unlimited capacities. Together with the use of commodity hardware in a large scale, providers are able to exploit the economies of scale. Furthermore, Cloud computing affords end customers an on-demand access as well as a high connectivity and a pay-as-you-go model for payment while just a little or even no commitment is required by them [1, 9].

Since Cloud computing became an increasingly applied technology for providers who are offering either software or hardware resources as a service over the internet

W. Abramowicz and A. Kokkinaki (Eds.): BIS 2014, LNBIP 176, pp. 206–216, 2014.

during the last years [6], Cloud computing is still a rapidly evolving paradigm where new developments incessantly emerge, especially in the open-source community [5]. All the benefits provided by Clouds are enabled by the abstraction of the Cloud infrastructure, more precisely computing resources, through the key technology of virtualization. The application of virtualized infrastructure in data centers will rise further on in the future. It enables a more efficient use of hardware through server consolidation. Additionally, virtualization affords a significant flexibility together with a number of new workflows which can reduce the total cost of ownership and improve the administrators' productivity. While providing those benefits, virtualization also introduces new challenges to operations management of data centers providing Clouds [28].

As operations management regards the efficient and effective management of processes that convert enterprise resources into a specific product while meeting consumers' as well as providers' requirements [13], Cloud operations management considers process management as well as resource management within a service-offering Cloud environment. The simplification and automation of processes are key enablers for a successful Cloud deployment [10]. Moreover, efficient Cloud resource management providing proper performance isolation, higher consolidation and elastic use of underlying hardware resources is a requirement for affordable Cloud operations management [23]. Adaptive resource management has the potential to increase the usability of Cloud applications while maximizing resource utilization which allows Cloud providers to manage resources more efficiently [16].

Besides the issue of management within a single Cloud, there is also a need for effective management of processes and resources across independent Clouds. The idea is that multiple providers are able to cooperate seamlessly to maximize their mutual benefit. Capabilities will be aggregated using federation concepts and interoperability to provide a seemingly infinite service computing utility. This approach enables to democratize the supply side of Cloud computing by allowing small and medium-sized enterprises (SMEs) as well as new entrants to become Cloud providers which encourages competition and innovation [22, 27]. Appropriate Cloud operations management should support these processes as well.

Although there are already solutions to deploy Clouds in practice yet, classification and comparability are hardly possible due to the fact that the field of Cloud computing is still evolving rapidly. In order to face prospective requirements to a Cloud, consolidation of the related work is necessary to support the vision of standardization of Cloud operations management. For this purpose, this paper analyses the current state of the art and demonstrates how Cloud operations management is considered in the literature.

2 Literature Review

This section reveals the state of the art in the context of Cloud operations management. Thereto, the applied methodology is depicted and the results are presented well-arranged. Afterwards, a classification is given to demonstrate the increasing relevance

of the field Cloud operations management and to make the determined articles comparable.

2.1 Research Design

For this paper, a systematic search for literature was conducted based on work by Seuring and Müller as well as Mayring and Brunner [20, 25, 26]. For this purpose, the established literature search engines Google Search, Google Scholar, SpringerLink, ACM Digital Library and IEEE Xplore were used. The applied search terms were "(cloud | (virtual | virtualized)) & (resource | process | infrastructure | operations) & management" and have been chosen based on a defined scope, namely "sets of approaches to support the operations in virtualized data centers". 48 documents have been identified. Furthermore, the results were filtered by that scope as well as by the relevance determined by the h-index. Finally 23 papers plus one book remained.

The determined literature is organized in a logical order. The scope of the contexts varied between operations management within one single Cloud and management of operations across different Clouds. The first category is presented initially, coming to the second one subsequently. Within these two categories, the literature is sorted ascending by the release date. Although all the information of the literature was useful to draw a picture, some articles had a rather weak usefulness while others had a stronger contribution to the topic Cloud operations management.

2.2 Single Cloud Focus

In [32] a detailed ontology of the Cloud to establish the knowledge domain of Cloud computing itself is proposed while a detailed description of the Cloud service stack is presented.

The authors of [5] describe experiences using four of the most advanced open-source VM-based Cloud management platforms and compare their architecture and different implementation of features which are appropriate according to particular use cases. This contribution was quite useful for the basic knowledge.

A design and an implementation of a dynamic resource management prototype solution based on Eucalyptus is suggested in [16]. For that purpose, the response time of service applications is monitored and the compute resources are scaled up adaptively. This approach is validated with an experimental evaluation. Although this paper provided essential basics, for the scope of this paper the contribution was of rather little help.

One paper addresses the issue of autonomic virtual resource management [30]. A two-level architecture which separates application-specific functions from a generic decision-making layer as well as algorithms for automated resource allocation are presented. Simulation experiments validate the architecture and algorithms and a prototype is to be implemented.

In [28], common management workflows are examined and their impact on the resource usage in data centers is assessed by examining data from real-world virtualized deployments. It is shown, that management workload has considerable network and

disk I/O requirements and also scales with the increasing computing power in the data center proved by explicit values. The arising management operations are described including their statistic occurrence. This paper was very conductive and presents lots of relevant information.

The article [7] presents a description how data centers manage their virtualized resources using pre-installed and pre-configured VMs to be deployed. The problem in practical scenarios is that pre-configured VMs cannot satisfy varying requirements of users. Moreover, they occupy a huge storage to provide a big variety of OS and software combinations for users. To face that problem an effective architecture for application deployment is presented. Here, user requirements are converted to the Open Virtualization Format (OVF) which is a hypervisor-neutral standard package format for Cloud deployment. It provides an open specification for the packaging and distribution of virtual applications composed of one or more VMs and facilitates the automated, secure management of appliances as a functional unit. The approach is evaluated by a case study.

Haase et al. present a very interesting paper that focuses on intelligent information management in enterprise Clouds [11]. An overall architecture that abstracts a vendor-specific representation is presented, the eCloudManager Ontology which is based on semantic technologies to enable an open, vendor-independent integration of heterogeneous data center resources. The contribution of this article to the field Cloud operations management was rather high.

The management of big data in Cloud infrastructures concerning the issue of high scalable data on the one hand and no scalable database systems on the other hand is considered by Agrawal et al. [1]. A system is discussed which provides efficient data management with varying degrees of consistency and scalability.

A Cloud middleware named CLEVER which provides both virtual infrastructure management services and suitable interfaces at the high-level management layer while granting scalability, modularity, flexibility and fault tolerance is presented in [29]. The high-level management layer enables the integration of public Cloud interfaces, contextualization, security and dynamic resource provisioning within the Cloud infrastructure. To validate the framework a prototype is presented and discussed.

In [10] techniques to provide large scale resource management in Cloud environments along with their pros and cons are discussed. This paper is quite beneficial for the reflection of this paper. Altogether, this contribution was quite helpful.

Three resource allocation algorithms for SaaS providers who want to minimize infrastructure cost and SLA violations are proposed by Wu et al. [31]. These algorithms enable providers to manage the dynamic change of customers, mapping customer requests to infrastructure level parameters and handling heterogeneity of VMs while considering QoS parameters like arrival rate and service initiation time. A simulation is used as evaluation study to show that costs can be lowered.

An efficient resource management solution for small and medium sized IaaS-Cloud providers with limited resources is presented in [12]. The goal is to better utilize their resources together with minimum operational costs by a well-designed underlying hardware infrastructure, an efficient resource scheduling algorithm and a set of

migrating operations of VMs. Use cases are described to explain the provider's day-to-day operation where resource allocation is carried out.

Hengxi et al. deal with queue management of incoming service requests [14]. The problem of Cloud resource management is discussed and an approach to determine the optimal capacity of a Cloud's resource pool is suggested based on the queue theory and the global optimization theory. This article was rather less beneficial.

2.3 Cross-Cloud Focus

In [18] an integrated Cloud computing stack architecture is shown to serve as a reference point for future mash-ups and comparative studies. The term Cloud ecosystem is introduced as the existing Cloud landscape and it is also presented how it maps into the proposed architecture. This article was quite helpful contribution.

The article [22] presents a new open-source federated Cloud model, the RESERVOIR Project which addresses the limited scalability of single-provider Clouds and the lack of interoperability among single Cloud providers. The RESERVOIR Project is initiated by the European Union and enables IaaS providers to dynamically partner with each other and create a seemingly infinite pool of IT resources while fully preserving their individual autonomy in making technological and business management decisions and keeping the quality high. This contribution was rather conductive.

Sotomayor et al. afford an overview of existing open-source solutions for virtual infrastructure management as well as of Cloud providers [27]. The Cloud ecosystem is introduced including the proposal to realize the cross-Cloud management through the mentioned RESERVOIR Project. Altogether, the information provided by this article was quite beneficial.

The authors of [8] introduce the issues of a so called system management which arises when end consumers use resources of more than one single provider. Since each provider exposes different interfaces to their compute resources utilizing different architectures and technologies this paper suggests an architecture to facilitate the management of compute resources from different Cloud providers while satisfying the Cloud paradigm. In this approach, a decision-making component decides when to provision and when to de-provision resources upon a cost-benefit analysis using the current state of compute resources. An implementation of this cross-Cloud system management architecture including an appropriate user interface is proposed. Empirical results obtained from experimentations show that the performance of the suggested architecture is independent from a given provider.

The state of the art in managing applications across multiple Clouds applying IBM Altocumulus Cloud middleware platform is demonstrated in [19]. It provides a uniform, service-oriented interface to deploy and manage applications in various Clouds and facilities to migrate instances across Clouds.

In [24] the usage of business-oriented Cloud governance to unify Cloud resource management model is discuss by Sedaghat et al. Besides the introduction of challenges for developing such a model a preliminary sketch is presented.

Sarkar et al describe a monitoring and event-based automated incident management system in a PaaS-Cloud in [23].

The authors in [3] introduce the design and implementation of the new KOALA Cloud management service which enables to manage hybrid Cloud resources. It allows to work with a large variety of services as well as public and private Cloud providers in a seamless and transparent way.

General tasks of Cloud management and presents the KOALA solution besides a number of other established Cloud management solutions explicitly is described by Baun et al. in [2]. A detailed overview is given including all available features and interfaces.

In [15] an architecture-based meta-model for the development of automated programs to manage platform facilities such as middleware and VMs is presented by the authors Huang et al.

Kecskemeti et al. discuss the issues of federated Cloud management like networking, specific VM management interfaces and storage interoperability [17]. An approach to extend federated Cloud management architectures with autonomous behavior is proposed. Knowledge management systems are used to facilitate the decision-making process of the classical monitoring-analysis-planning-execution loop.

2.4 Classification of the Determined Articles

Now that contents of the papers found are described, this section constitutes an attempt to classify the papers to make them comparable. The identified literature can be distinguished into three dimensions: scope, result and year. The corresponding document's scope is distinguished between operations management in a single Cloud and operations management in a cross-Cloud environment. The result represents the type of output the document generates, as model, prototype, approach and overview are possible values. A model means a description of an implementable artifact while a prototype depicts a specific, elementary software solution. Furthermore, an approach describes best practice methods to face operations management and an overview clarifies its context. The year corresponds the document‘s year of publication.

Figure 1 shows a bar chart that counts the examined documents for each regarded year distinguished by their scope. The year 2012 is seemingly under-represented what is probably justified by the fact that the literature databases are not up to date yet. Regarding this, it becomes clear that the topic is quite relevant since 2009 with a relatively constant number of documents each year. Before, there was only an insignificant consideration of the superior topic Cloud computing in the literature as the term was defined first by the National Institute of Standards and Technology (NIST) consistently in 2009 [21]. Figure 2 depicts another bar chart that connects the documents' scope with their results. Most documents deal with a specific model proposal for single Cloud environments or present approaches for cross-Cloud environments. Only a few consider specific prototypes or overviews. This demonstrates the lack of holistic consideration of Cloud operations management once again. Nevertheless, the total count of 23 papers plus one book is fairly small at all which can be interpreted as the topic is still not elaborated entirely.

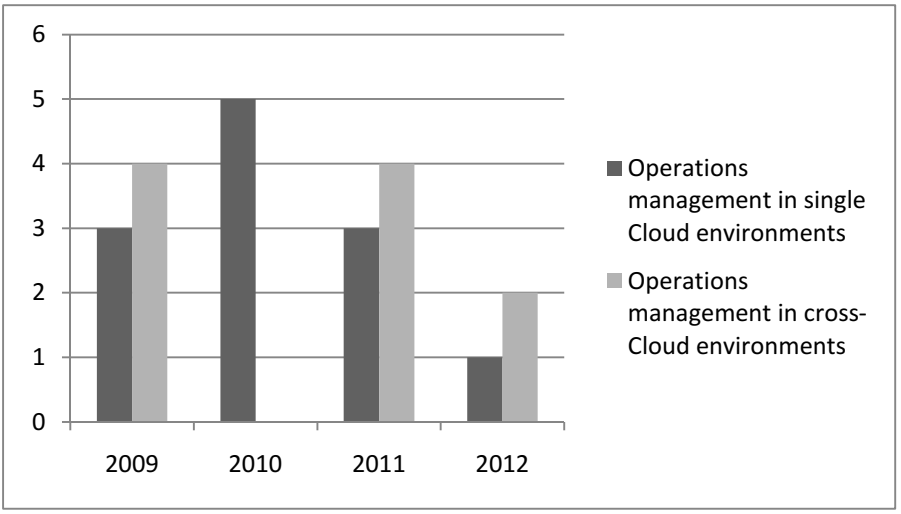


Fig. 1. Count of examined papers in relation to their publication date

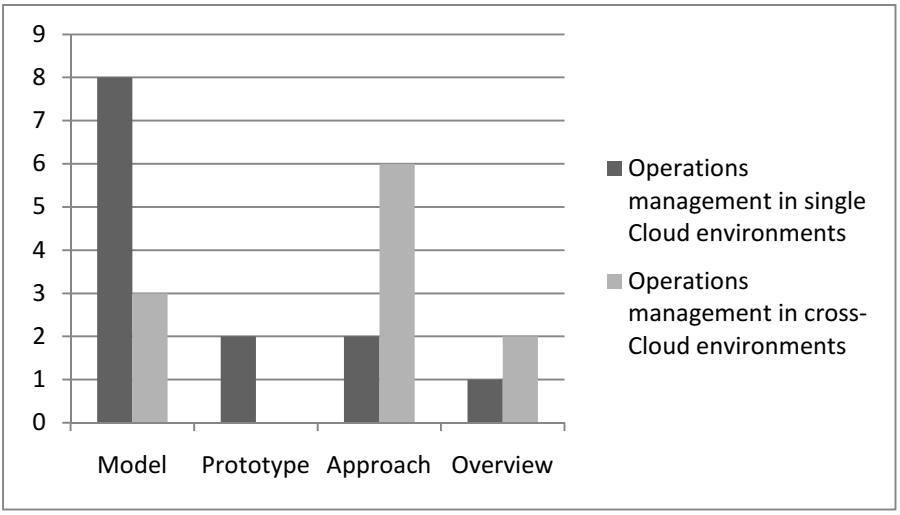


Fig. 2. Count of examined papers in relation to their result

3 Discussion

In this section the determined information of the literature review is consolidated and a common picture of Cloud operations management is deduced. The literature showed that Cloud operations management owns a stack consisting of two layers, a high-level management layer which includes process management and a virtual infrastructure layer which contains resource management [29] which is pictured in Figure 3.

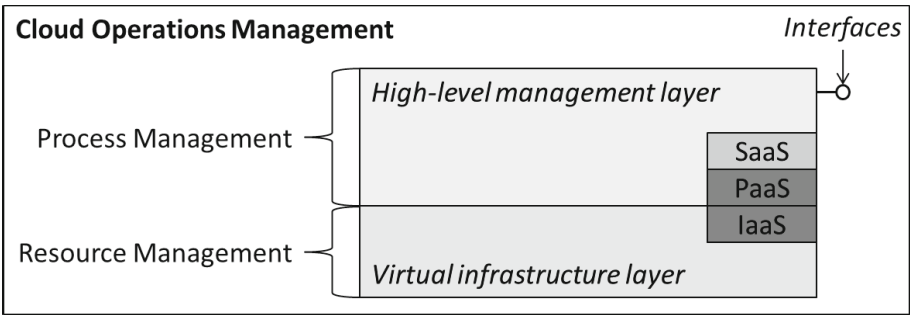


Fig. 3. Cloud operations management stack

About the half of the articles deal with cross-Cloud domains depicted as interfaces of the process management layer to other Clouds. Among others, process management includes the SaaS and PaaS layers of the Cloud stack. Here, the lifecycle of virtual deployments is considered to intend a closed loop composed of four phases. In detail, these are a constantly monitoring phase complemented by an analysis phase that examines and evaluates monitored data, a scheduling phase which applies processes that are pre-defined in a portfolio to an occurring issue and an execution phase that implements the selected process (e.g. by providing additional space) [2]. Additionally, process management contains certain administration tasks [28], namely

- Manage periodic snapshots to keep VM configurations revertible,
- Perform host and VM patching to keep the environment up to date,
- Handle boot storms which occur depending on specific daytimes as customers start to use a service increasingly,
- Manage an automated live migration of resources to realize a load balancing,
- Perform an after-hours maintenance where operation tasks like testing or committing VM snapshots are conducted.

These tasks generate network workloads which can stress the infrastructure more than the load of allocated resources itself while these operations are time critical. This overhead should be considered by administrators by factoring the resource usage within a Cloud infrastructure [28]. Process management should include external Cloud interfaces, security management to facilitate security as well as protection of VMs and service level management to ensure the adherence of SLAs, too. In order to realize that, an approach to insert a layer into the Cloud stack, is suggested in the literature. Here, both aspects, the agreement on the quality of service and service monitoring at runtime, are considered.

Resource management in a Cloud environment contains the IaaS layer of the Cloud stack considering the deployment of resources and the operations connected with them [2, 8, 29]. These are particularly

- Network management to manage IP addresses and the creation of static networks,
- Cluster management to manage provision groups to assign VMs to custom clusters or hosts,
- Monitoring which constantly examines the state of the Cloud infrastructure,
- Control which includes operations as launch of VMs, restart of VMs, scaling of VMs, migration, I/O processing on VMs (like file uploads), shutdown of VMs, viewing VMs' logs and accessing VMs' consoles.

In order to achieve adaptive resource management a dynamic real time allocation is required. For this purpose load balancing algorithms are necessary [12, 16] realized by the high-level management layer.

The literature review made clear that there are already efforts that consider operations management in Cloud environments. However, the common state of this context is not consolidated yet, because different scopes exist. Specific interfaces of different Cloud infrastructure providers lead to a variety of individual solutions. Some approaches, like the KOALA or the RESERVOIR Project, try to meet the demands on Cloud operations management in a holistic manner, but they still do not cover all aspects of Cloud operations management completely since they are still in their early stages. There are deficits with the support of generic infrastructure providers, monitoring and on-demand allocation of VMs. Furthermore, the Cloud market currently still lacks generic management solutions with good usability [3]. Existing approaches either provide poor management controls, low consolidation ratios or do not scale well [10].

To provide appropriate Cloud operations management, some remaining issues still have to be discussed prospectively. There is a need to update some common data structures in a short span of time requiring a good decomposition of tasks and fine grained locking of virtualized resources. In addition, heterogeneous clusters need to be regarded since VMs may be compatible only with a few hosts that do not have enough capacity to satisfy all the reservations. It is also important for the system to keep providing a low latency as the cluster size increases regarding the rising frequency of operations. Furthermore, resistance to failures is to be considered as the scale increases, the likelihood of hardware failures and failures of the resource management component increases as well [10]. Finally, the full lifecycle of VMs including the management of their resource requirements needs to be set up dynamically and automatically regarding specific enterprise policies like high availability or minimization of power usages [27].

4 Conclusion

This paper regards a reflection of the context of Cloud operations management. Withal, tasks and requirements of Cloud operations management were registered. For this purpose, a structured literature review was conducted to clarify how much this topic is already represented in the literature. Papers were examined, filtered by relevance and finally classified. A research gap could be identified since Cloud operations

management is a quite new field. Many approaches and solutions already exist, but a consolidation to a holistic reflection of Cloud operations management could not be identified.

Therefore, further research in this context is necessary to encourage the work within the field of Cloud operations management. Based on this paper, in future considerations a proposal for a holistic and generic description of the context Cloud operations management will be presented.

References

1. Agrawal, D., El Abbadi, A., Antony, S., Das, S.: Data Management Challenges in Cloud Computing Infrastructures. In: Kikuchi, S., Sachdeva, S., Bhalla, S. (eds.) DNIS 2010. LNCS, vol. 5999, pp. 1–10. Springer, Heidelberg (2010)
2. Baun, C., et al.: Cloud Computing: Web-Based Dynamic IT Services. Springer, Heidelberg (2011)
3. Baun, C., et al.: The KOALA Cloud Manager: Cloud Service Management the Easy Way. In: Liu, L., Parashar, M. (eds.) 4th IEEE International Conference on Cloud Computing, pp. 744–745. IEEE, Washington DC (2011)
4. Buyya, R., et al.: Cloud Computing and Emerging IT Platforms: Vision, Hype, and Reality for Delivering Computing as the 5th Utility. Future Generation Computer Systems 25, 599–616 (2009)
5. Cerbelaud, D., et al.: Opening the Clouds: Qualitative Overview of the State-of-the-Art Open Source VM-Based Cloud Management Platforms. In: Douglis, F. (ed.) 10th ACM/IFIP/USENIX International Conference on Middleware, p. 22. ACM, Urbana Champaign (2009)
6. Chen, X., et al.: A Model-Based Framework for Platform Management in Cloud. In: Ferreira, P., et al. (eds.) 13th International Conference on Middleware, p. 8. ACM, Montreal (2012)
7. Dastjerdi, A.V., et al.: An Effective Architecture for Automated Appliance Management System Applying Ontology-Based Cloud Discovery. In: 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing, pp. 104–112. IEEE, Melbourne (2010)
8. Dodda, R.T., Smith, C., van Moorsel, A.: An Architecture for Cross-Cloud System Management. In: Ranka, S., et al. (eds.) IC3 2009. CCIS, vol. 40, pp. 556–567. Springer, Heidelberg (2009)
9. Durkee, D.: Why Cloud Computing Will Never Be Free. ACM Queue 8, 20 (2010)
10. Gulati, A., et al.: Cloud-Scale Resource Management: Challenges and Techniques. VMware, Inc. (2011)
11. Haase, P., Mathäß, T., Schmidt, M., Eberhart, A., Walther, U.: Semantic Technologies for Enterprise Cloud Management. In: Patel-Schneider, P.F., Pan, Y., Hitzler, P., Mika, P., Zhang, L., Pan, J.Z., Horrocks, I., Glimm, B., et al. (eds.) ISWC 2010, Part II. LNCS, vol. 6497, pp. 98–113. Springer, Heidelberg (2010)
12. He, S., et al.: Real Time Elastic Cloud Management for Limited Resources. In: Liu, L., Parashar, M. (eds.) 4th IEEE International Conference on Cloud Computing, pp. 622–629. IEEE, Washington, DC (2011)
13. Heizer, J., Render, B.: Operations Management. Pearson Education (2011)
14. Hengxi, Z., Chunlin, L., Zhengjun, S., Xiaoqing, Z.: Resource Pool-Oriented Resource Management for Cloud Computing. In: Zhu, M. (ed.) Business, Economics, and Financial Sci., Manag. AISC, vol. 143, pp. 829–832. Springer, Heidelberg (2012)

15. Huang, G., et al.: Towards Architecture-Based Management of Platforms in the Cloud. Frontiers of Computer Science 6, 388–397 (2012)
16. Iqbal, W., Dailey, M., Carrera, D.: SLA-Driven Adaptive Resource Management for Web Applications on a Heterogeneous Compute Cloud. In: Jaatun, M.G., Zhao, G., Rong, C., et al. (eds.) Cloud Computing. LNCS, vol. 5931, pp. 243–253. Springer, Heidelberg (2009)
17. Kecskemeti, G., et al.: Facilitating Self-Adaptable Inter-Cloud Management. In: Stotzka, R., et al. (eds.) 20th Euromicro International Conference on Parallel, Distributed and Network-Based Processing, pp. 575–582. IEEE, Garching (2012)
18. Lenk, A., et al.: What's Inside the Cloud? An Architectural Map of the Cloud Landscape. In: ICSE Workshop on Software Engineering Challenges of Cloud Computing, pp. 23–31 (2009)
19. Maximilien, E.M., et al.: IBM Altocumulus: A Cross-Cloud Middleware and Platform. In: Arora, S., Leavens, G.T. (eds.) 24th ACM SIGPLAN Conference Companion on Object Oriented Programming Systems, Languages and Applications, pp. 805–806. ACM, Orlando (2009)
20. Mayring, P., Brunner, E.: Qualitative Inhaltsanalyse. Qualitative Marktforschung, pp. 669–680. Springer (2009)
21. Mell, P., Grance, T.: The NIST Definition of Cloud Computing. National Institute of Standards and Technology (NIST) (2011)
22. Rochwerger, B., et al.: The Reservoir Model and Architecture for Open Federated Cloud Computing. IBM Journal of Research and Development 53, 535–545 (2009)
23. Sarkar, S.R., et al.: Automated Incident Management for a Platform-as-a-Service Cloud. In: 11th USENIX Conference on Hot Topics in Management of Internet, Cloud and Enterprise Networks and Services, pp. 5–10. USENIX Association Berkeley, Boston (2011)
24. Sedaghat, M., et al.: Unifying Cloud Management: Towards Overall Governance of Business Level Objectives. In: 11th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, pp. 591–597. IEEE, Newport Beach (2011)
25. Seuring, S., et al.: Conducting a Literature Review - The Example of Sustainability in Supply Chains. In: Research Methodologies in Supply Chain Management, pp. 91–106. Springer (2005)
26. Seuring, S., Müller, M.: From a literature review to a conceptual framework for sustainable supply chain management. Journal of Cleaner Production 16, 1699–1710 (2008)
27. Sotomayor, B., et al.: An Open Source Solution for Virtual Infrastructure Management in Private and Hybrid Clouds. IEEE Internet Computing 13, 14–22 (2009)
28. Soundararajan, V., Anderson, J.M.: The Impact of Management Operations on the Virtualized Datacenter. In: Seznec, A., et al. (eds.) 37th Annual International Symposium on Computer Architecture, pp. 326–337. ACM, Saint-Malo (2010)
29. Tusa, F., et al.: CLEVER: A cloud-enabled virtual environment. In: IEEE Symposium on Computers and Communications, pp. 477–482. IEEE, Riccione (2010)
30. Van, H.N., et al.: SLA-Aware Virtual Resource Management for Cloud Infrastructures. In: 9th IEEE International Conference on Computer and Information Technology, pp. 357–362. IEEE Computer Society, Xiamen (2009)
31. Wu, L., et al.: SLA-Based Resource Allocation for Software as a Service Provider (SaaS) in Cloud Computing Environments. In: 11th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, pp. 195–204. IEEE, Newport Beach (2011)
32. Youseff, L., et al.: Toward a Unified Ontology of Cloud Computing. In: Grid Computing Environments Workshop, pp. 1–10. IEEE, Austin (2008)

Resource Mining: Applying Process Mining to Resource-Oriented Systems*

Andrzej Stroiński, Dariusz Dwornikowski, and Jerzy Brzeziński

Institute of Computing Science, Poznań University of Technology
Piotrowo 2, 60-965 Poznań, Poland
{Andrzej.Stroinski,Dariusz.Dwornikowski,Jerzy.Brzezinski}@cs.put.poznan.pl

Abstract Service Oriented Architecture is an increasingly popular approach to implement complex distributed systems. It enables implementing complex functionality just by composing simple services into so called business processes. Unfortunately, such composition of services may lead to some incorrect system behavior. In order discover such depreciances and fix them, process mining methods may be used. Unfortunately, the current state of the art focuses only on SOAP-based Web Services leaving RESTful Web Service (resource-oriented) unsupported. In this article the relevance of adapting the Web Service Mining methods to new resource-oriented domain is introduced with initial work on process discovery in such systems.

Keywords: process mining, business process, logging, SOA, REST.

1 Introduction

Currently, often used approach to the implementation of distributed systems is Service Oriented Architecture (SOA). This approach reduce costs of development, maintenance and provides an easy integration of system implemented accordingly to it. It is possible by splitting simple system functionalities into independently developed applications called Web Services (WS). Later on, composition of many WS into business processes is used to provide more complex functionality.

Nowadays, two different approaches to SOA are widely recognized [20]. The first one are SOAP-based WS, which are highly standardized, and use WSDL (Web Services Description Language) to describe their procedural interfaces and rely on SOAP (Simple Object Access Protocol) as their communication protocol. The second approach, introduced in [11] is REST (Representational State Transfer) and RESTful (resource-oriented) WS, which take a declarative approach, are based on resources rather than functions [21].

In both of the approaches however, the same problems with composition may yield incorrect system behavior, i.e. deadlock, livelock. In addition, number of

* This work was supported by the Polish National Science Center under Grant No. DEC-2012/05/N/ST6/03051.

W. Abramowicz and A. Kokkinaki (Eds.): BIS 2014, LNBIP 176, pp. 217–228, 2014.

composed and invoked services during system execution may be tremendous, making it hard to manage. In order to deal with this problem, a research of process mining (PM) [2] may be used. Up to date, a lot of work has been already done concerning: log extraction [15,19], process model discovery [1,13,3], conformance checking and enhancement [5].

Being a prominent and fast developing research area, PM has been also applied to SOA, raising a new problems and challenges [12] that need to be address like: cross-organization PM [8], event data preprocessing [2], process models discovery from SOA services logs [9,10,6], improving WS behavior [4] and gathering logs [16]. As it can be seen process discovery, and generally PM, has been only applied to SOAP-based WS SOA systems. We believe that REST systems could also benefit from applying PM techniques. For that to be possible, one first needs to gather logs from a system, which are always the first step in every PM method. There are papers that deal with gathering and collecting logs from SOA systems in order to apply PM techniques, or Web Service mining techniques. In [16] Authors tackle with the problem collecting event logs in order to extract process traces from application systems and integration portal log files. In [7] and [18] methods to deal with correlation of events with processes and processes instances are presented.

Unfortunately, authors are considering only the interactions between services without taking into account local events. Furthermore, all the articles focus on SOAP-based systems, so the results they present cannot be directly applied to resource-oriented systems (ROS, consisting of RESTful WS), due to different nature of SOAP-WS and REST.

In this article we tackle the problem of adapting the Web Service Mining methods to RESTFul WS domain. In addition, we introduce context logging, a technique of log enrichment in order to make possible to infer process related data in ROS (Sec. 2). Furthermore, process and process instance reconstruction algorithm for ROS, based on approaches for SOAP-based WS is presented (Sec. 2.1). We also propose and discuss a prototype framework implementation (Sec. 3). Finally, utilization of proposed methods with classic PM methods in order to achieve Resource Mining (RESTful Web Service Mining) is shown (Sec. 4).

2 Resource Mining: The Resource-Oriented Approach to Web Service Mining

In order to discover process models in ROS there is a need to adopt the already existing methods of Web Service Mining and/or develop new ones to respect differences in ROS features in contrast to SOAP-WS:

1. A service is only an application component composed of a callable set of resources, which are important from a client's perspective. Therefore, only individual resources need to be considered.
2. A service execution state is stored on a client's side (active), not on a server side (resources). Thus, a process logic is executed in a client by executing a predefined, finite set of CRUD operations on resources.

3. Resources are passive, they only provide data representation and implementation of a client callable operations. This approach introduces stateless communication, and unified interface [11].
4. Business process is a resource. Complex functionality in ROS is achieved by composing system resources invocations into workflows or business processes. Upon client's action, a passive resource may, on behalf of that client, act as client for other resources, we call it a *process resource*.
5. Resources are hierarchically dependent on each other, some of resource representations may be included in other resource representations. Correctly modeled and implemented ROS will use `URI`s to pinpoint such inclusion [21].
6. HTTP protocol is used as communication layer (in SOAP-WS it only serves as one of transport layers to ensure SOAP messages delivery), so the semantics of HTTP messages drives request handling.
7. HTTP guarantees receiving response for every request. We assume only synchronous communication as a basis for further discussion about more complex communication patterns (sequence of synchronous interactions modeling asynchronous communication).
8. In contrary to SOAP, HTTP lacks one-way communication so there is no such type of communication in ROS (standard request-response model).
9. There are no standards like WS-Addressing or WS-Conversation, so there is no support for using and logging process related information like process IDs and process instance IDs, so the correlation patterns from [7] are hard to fulfill and solutions like [18] are not sufficient for ROS.
10. Process resources may be nested, and may be further orchestrated into complex process resources. In such a case, process logic is also nested in internal events of resources. Consequently, there is a need to discover not only traces of communication events and correlate them into process instances but also internal resource events.

The crucial problem in log collecting in ROS is the lack of appropriate logging level available in the current SOA implementations [10]. This problem occurs due to the usage of application servers developed exclusively for request-response model of interaction. In this model, server passively awaits for requests, upon receiving it, it is processed and a response is sent back to the client. Hence, only information about received message and returned response is stored in an event log. What is lacking is process related data: process instance id and process id. In addition, resources may act as clients and such events are usually not stored in log (difference (4)).

Next, there is a need to group events concerning each of the resources belonging to the particular RESTful service (difference (1)). Usually, services store invoked URI address in log so this information may be used, or if application server does not support such feature, a solution is presented in Sec. 3. Next, there is also a problem of handling resources by parallel instances. Current application servers like Apache Tomcat create new instance of resource for each incoming request to the same URI. In consequence, each resource instance, logs information concurrently into a log file, so event log interleaving problem occurs (Fig. 1).

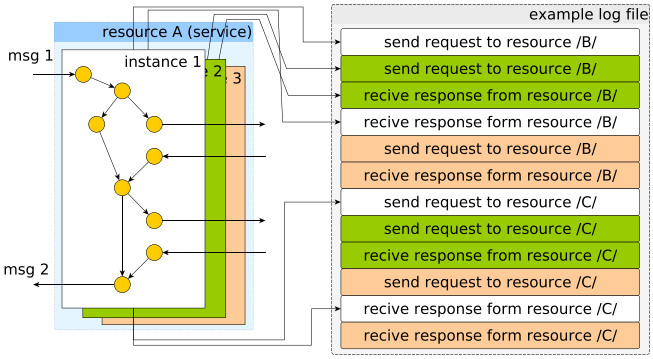


Fig. 1. log interleaving problem

`Instance 1` of `Resource A` is invoked and sends a request to `resource B`, an appropriate log entry in log is stored. Next, `instance 2` is created upon second request to `resource A`. The `instance 2` request to `resource B` is sent, and response for that request is received (line 2 and 3 at Fig. 1). Next, `instance 1` receives response and logs this information (line 4). As the example shows, if there is no information about instances in the log, that create log entry, there is no possibility to tie the receiving of message to sending it. In addition, log also includes information about incoming messages like `msg 1` and returned messages `msg 2`. There is a need to correlate the messages with each other and with outgoing messages, in order to associate them with proper service instances, as well as to keep log ordering relation [3]. This allows to discover local process of each resource (Fig. 2b):

$$a \succ_L b \text{ iff there is trace where event } b \text{ immediately precedes event } a \tag{1}$$

This relation orders all local events of some resource (local process at Fig. 2b). In order to deal with above problems we introduce context logging. The main concept is to add a unique ID (Context ID) of the resource instance to each logged event. As a result each service instance will add additional field to event log during logging called `context`. This context simply correlates incoming messages, with outgoing messages, and some local events. Such a context log allows to specify events that take place within different instances of resources allowing to generate an independent event log files for each resource in the service, and each instance of that resource (local log at Fig. 2b). In addition, if we enforce adding local context as a additional HTTP header (it is possible because of difference (6)) it is also possible to correctly preserve ordering (correlation) relation introduced in [18] (*atomic correlation condition*) or in [7] (*reference-based correlation*) between interacting resources (`res`) based on context information in HTTP header (Eq. 2).

$$
\begin{aligned}
a \succ_{ctx} b \text{ iff there is trace in event log where } & \#_{ctx}(a) = \#_{hctx}(b) \wedge \ \#_{res}(a) = resA \\
& \wedge \#_{res}(b) = resB \wedge resA, resB \in Res \wedge resA \neq resB \wedge \#_{destURI}(a) = resB \\
& \wedge \#_{srcURI}(b) = resA, \text{ where } Res \text{ is a set of all resources in the system,} \\
& \text{and } \#_{ctx}(e) = A \text{ means value of field } ctx \text{ of event } e \text{ is A}
\end{aligned} \tag{2}
$$

These relations describe a situation where `resA` invokes `resB` ($\#_{destURI}(a) = resB \wedge \#_{srcURI}(b) = resA$) and logs this information with $\#_{ctx}(a)$ label as event a and `resB` receives this message and logs this event with context label passed by `resA` ($\#_{hctx}(b) = \#_{ctx}(a)$) and with local context label $\#_{ctx}(b)$. Next step is to reconstruct session and a global process and generate appropriate `processID` and `instanceID`, basing on context information (session reconstruction and global process in Fig. 2b). In order to reconstruct session one needs to apply information about which resource instance invokes other resource instance. Such information allows to retrieve the whole workflow information of interacting resources during business process execution. In order to achieve that, we ask each resource to send its local context in HTTP header to its callees. Then each callee needs to log this header as a receiving event log entry with its own local context. As a result, each of invoked service has information about local context within it was called. Based on the context ordering relation and partial context information, the algorithm for session reconstruction can be applied.

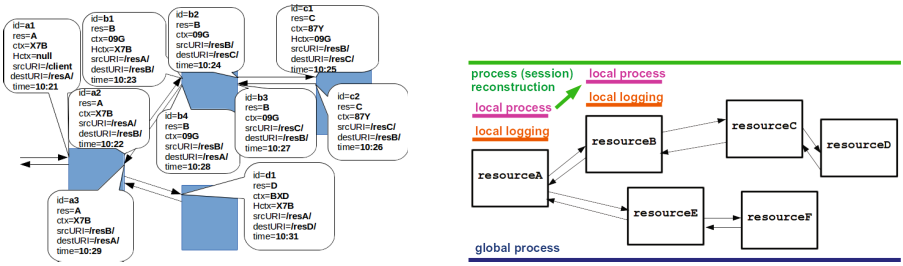
2.1 Simple Process and Instance Reconstruction Algorithm for Resource-Oriented Systems

The main idea behind session reconstruction is to add appropriate `processID` and `instanceID` to events in the log. The main problem is to tie events with a process instance, i.e. process run. In our approach we are using the idea of context logging from Sec. 2. We are assuming that each resource enriches its log entries with context field generated by each of its instances. This allows us to distinguish event log entries created by different resource instances (even if they are in the same log file). Next, if resource plays client role during process execution, it must include context information into its all outgoing messages. This ensure that context information is transfered to nearest neighbors, and allows to tie interacting resource instances with each other. We use HTTP header (`hctx`) to transfer context. The example of log entries generated by resources during interaction is presented in The Fig. 2a.

Therefore, based on the relation in eq. 2, and the obtained context log, we are construct a chain of connected resource instances. First, we generate a set `CTX` that contains information about the dependence among resource instances occurring in the log:

$$CTX = \{(e_1, e_2) \mid e_1 \in L \wedge e_2 \in L \wedge (e_1 \succ_{ctx} e_2)\} \tag{3}$$

We are looking a pair of events in the event log that represent communication between two resource. Such events in resource-oriented (RESTful) log are easy



(a) Log entries example (b) Idea and limitation

Fig. 2. Context logging

to find because they include additional data related to HTTP protocol (header, method etc.). The sender of message will include its local context into `hctx` header of message, so as a result the receiver will log in event e_2 the sender local context (it is included in HTTP header) next to its own local context. Into set CTX we put tuples of events between communication resources, where local context of first event (e_1) is equals to received from communication invoker context in event e_2. Next, we search the log for global process starting events (communication event sent by process principal), according to:

$$F = \{f \mid f \in L \wedge \#_{hctx}(f) = null\} \tag{4}$$

Global process starting event is recognized be empty `hctx` HTTP header value ($\#_{hctx}(f) = null$). This occurs when $\#_{res}(f)$ resource is invoked by a process principal, because process principal is not a part of process so it does not include context information in invoke messages. Each event in F represents the initial event in global process execution so the size of set $|F|$ represents a number of process instances occurring in event log. Next, all so called context chains of events are calculated (based at correlation condition in [18]). The context chain is an ordered set of events that represents context flow during process execution in one global process instance (one chain represents one global process instance).

foreach $f_j \in F$ **do** $CHAIN_j = \{f_j, E_j, CTX_j\}$, where f_j is a starting event in this chain , and E_j is a set of events in this j-th chain and$CTX_j$is a set of context dependency between events in $E_j \cup \{f\}$, where $j = 1...|F|(|F|$ is a number of process instances occurring in log). (5)

In order to do that, we need to find all events sets of context dependent events in each of the context chains (one context chain for each starting event):

$$E_j = \{e \mid (e \in L \wedge \exists_{e' \in L \wedge e' \in E_j}(e \succ_{ctx} e' \vee e' \succ_{ctx} e))\}, \text{ where } (\exists_{e'' \in E_j}\ f_j \succ_{ctx} e'') \tag{6}$$

Set E contains events e, such that all events in this set are context dependent on at least one other event in this set, additionally at least one event from this set is context dependent on starting event f_j. Next, the set of context dependencies (CTX_j) between events of set E_j is calculated as follows:

$$CTX_j = \{(e_1, e_2) \mid e_1 \in E_j \wedge e_2 \in E_j \wedge e_1 \succ_{ctx} e_2\} \tag{7}$$

Set CTX consists of tuples (e_1, e_2) where event e_2 is context dependent on event e_1. In consequence, each context chain shows mutually interacting process instances in some (still unknown) global process. As a result, the process `instanceID` may be generated and added to each of context chains. Further, each event can obtain `instanceID` from context chain it belongs to. Unlike most of approaches, other events (e_l) – not only invocation events, must be added in order to take local processing of resources under consideration (differences (10)). This results in more accurate process models because sending messages may be dependent on some local resource event. This results in the $Instance$ set:

$$Instance_j = \{(f_j, E_j \cup \{e_l \mid e_l \in L \wedge \exists_{e' \in E_j} \#_{ctx}(e_l) = \#_{ctx}(e')\}, CTX_j, SUCCj)\} \text{ , where } \\ f_j \text{ is a starting event in chain } CHAIN_j \text{ and } E_j \text{ is a set of events in chain } \\ CHAIN_j, \text{ and } CTX_j \text{ is a set of context dependency occurring in this, } \\ \text{process instance and } SUCC_j \text{ is a local resource events ordering set} \tag{8}$$

$SUCC_j$ is a set of tuples showing local order relation among events of the same resource instance. It contain all the events belonging to the resources (instances) involved in $j-th$ context chain.

$$SUCC_j = \{(e_1, e_2) \mid e_1 \in L \wedge e_2 \in L \wedge \exists_{e' \in E_j} (\#_{ctx}(e_1) = \#_{ctx}(e_2) = \#_{ctx}(e')) \\ \wedge (\#_{res}(e_1) = \#_{res}(e_2)) \wedge (e_1 \succ_L e_2) \wedge (e_1 \neq e_2)\} \tag{9}$$

The final step is to determine which of the found instances belong to which process. The idea to discover processes, and correlate instances with them is based on differences (1) and (4) that everything is represented in form of resource (even business process). In ROS, business processes are executed by invoking resources call other resources on behalf of the *process principal*. Therefore, the final step is to analyze resource property of each first log entry $(\#_{res}(f_j))$ of each of $Insance_j$ in order to find such process resources. We are analyzing only the first event in each instance, as they are invoked by process principals $(\#_{hctx}(f) = null)$, so they are the starting point of process execution. Next, for each unique resource (called *process resources*) we are generate `processID`, because each instance starting from the same resource is an instance of the same process resource. This allows us to correlate instances with process by calculating sets of processes instances for each of process resources occurring in the log:

$$Process_n = \{i \mid i \in AllInstaces \wedge \exists_{i' \in Process_n} ((\#_{res} first(i) = \#_{res} first(i') \\ \wedge i \neq i') \oplus i{=}i')\}, \text{ where function } first() \text{ returns } f_j \text{ for } CHAIN_j \tag{10}$$

As a result the algorithm returns a set of $Process_n$ sets that include several $Instance_j$. Based on this, there is a need to review all events in event logs of all resources in the system, and add to them instanceID accordingly to ID of $Instance_j$ that this event belongs to. Then add `processID` accordingly to ID of $Process_n$ to which that event $Instance_j$ belongs to. As presented in Fig. 2a there is only one global process $(Process_0 = \{Instance_0\})$ and one instance $(Instance_0 = \{\{a1\}, E_0, CTX_0, SUCC_0\}$, where $E_0 = \{a2, a3, a4, b1, b2, b3, b4, c1, c2, d1, d2\}, CTX_0 = \{(a2, b1), (b2, c1), (a3, d1)\}$, and $SUCC_0 = \{(a1, a2), (a2, a3) \ldots (d1, d2)\})$

3 Non-invasive Context Logging for JSR-311 with AspectJ: A Case Study

Context logging can be used to differentiate among separate resource instances in separate process instances of multiple processes. The idea behind context logging is to inject HTTP headers into messages that pass through the system. This simple technique can be implemented in three ways: service instrumentation, proxy servers introduction, and semi non-invasive way.

We show that non-invasive logging is possible for a wide range of enterprise systems, i.e. Java based RESTFul-WS, implemented according to the JSR-311 standard [14]. Here we use AspectJ [17] (Aspect Oriented Programming paradigm, AOP) and Apache Tomcat application server.

Jersey comprises to JAX-RS, a JSR-311 standardized API of implementing RESTful-WS in Java. Both, the standard, and Jersey are widely used in a number of enterprise application servers and frameworks. We present a proof of concept implementation of our AspectJ context logger for RESTful systems implemented according to JSR-311, and in fact, this is our only technical requirement. Our approach will work for any Java application server and implementation of JAX-RS. We believe that the same approach can be used for any other technology that offers support for AOP, such as .NET, Python or Ruby.

We take advantage of the fact that in JAX-RS (Jersey), every RESTful Web service needs to be defined in a class, annotated with certain decorators, e.g. `@Path`. The listing below shows a simple Web service implemented in Jersey. `@Path("/hello")` says that the Web service will be accessible under `"/hello"` URI resource. The method annotated with `@GET` and `@Path` handles every GET operation issued on `"/hello/world"` resource, `@GET` can be substituted with any other HTTP operation. `@Produces` or `@Consumes` in the case of POST, PUT defines content type the resource returns or accepts.

```
@Path("/hello") // every class has @Path
public class Hello {
   // every method has @OP annotation
   @Produces("text/html") @GET @Path("/world")
   public Resource handler(@Context HttpHeader headers,
        @Context HttpServletRequest request) {}
```

Thanks to the standardized API, AspectJ context logger for incoming messages can be implemented in a simple way. One needs only to define pointcut, which catches every execution of any method placed in any class annotated with `@Path`, `@Produces` and `@GET`. In our implementation an advice is called when the pointcut is reached. First local context is generated, which in our case is the hash-code of a current object instance. Next, if the `HCTX` header is present in the request, it is stored and logged alongside with the local context, remote caller IP, HCTX value, and request URI from the `@Path`. Local context is then appended to every outgoing request from the current service instance in a `HCTX` header.

```
execution(@javax.ws.rs.GET @javax.ws.rs.Produces public * *(..)) &&
          within(@javax.ws.rs.Path *)
```

The situation gets more complicated when we want to catch and log messages sent from a service. In that case, not only we have to alter the outgoing message

with context logging HTTP header `HCTX` but also the HTTP client call may be done in some arbitrary way. In our case, we assume that these external calls are done from the same thread that handles the incoming request, i.e. synchronously. We also assume that JSR-311 client API is used. Therefore, a pointcut can be defined to catch all calls to methods named `request*` within classes annotated with `@Path`. Such an approach allows us to alter the request headers originating from the service, and thus pass the context to external services, according to context logging approach. Assuming that external services are also equipped with our aspect context logger, logs of all messages received and sent in the whole system can be created.

```
call(public * *.request(..)) && within(@javax.ws.rs.Path *)
```

There are some requirements we need to impose on how services are implemented with Jersey. We require that every method handling requests needs to return `Resource` object, and take the following arguments: `@Context HttpHeader` and `@Context HttpServletRequest`. This is needed to extract information, such as remote caller IP, request headers, and to inject our own header. Another difficulty we came across is the way the situation when service we equip with aspects calls some external service. In our approach we assumed that the call is done in the same thread as incoming request handling but this does not need to be the case. If it's not the case, it is still possible to implement a logger, by examining the call stack to determine how the current thread was called. By comparing this with the call list kept in aspects, it is possible to determine which service instance was the original caller.

4 Applying Alpha Algorithm for Resource-Oriented Context Preprocessed Log

After preprocessing the context log by the algorithm in Sec. 2.1, it is now possible to apply classic PM algorithms. As it is shown in Sec. 1, there are a lot of process model discovery approaches available up-to-date, but in our initial work on service mining in ROS, we have used basic alpha algorithm (AA) [3]. The reason for this is to show on simple and representative example that PM is applicable but some additional work is still need to be done. The idea behind AA is to use ordering relation (Eq. 1) that occurs in the log file to discover process model in form of a petri net. Unfortunately, in the real case scenario in ROS, it is unlikely that each resource in the system will log into the same log file with respect to some global clock and with respect to some global ordering relation. Therefore the problem of gathering logs from distributed resources with respect to global ordering arises. In addition, the basic version of AA does not take the resource perspective into account. So the first step is to make logs unique globally (usually events IDs are only unique locally at resource). Without distinction of resources, two events occurring at different resources may leave identical log entries, so as a global identifier is the concatenation of resource URI (is unique by the definition) and its local identifier (unique at the resource).

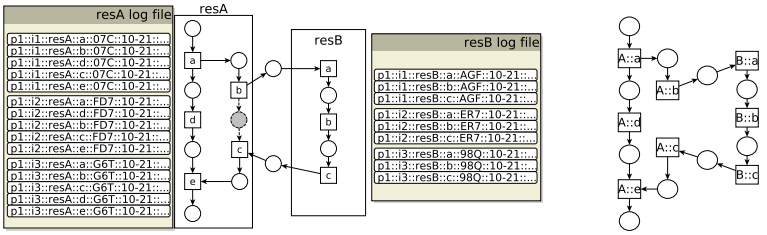


Fig. 3. a(left) – original process model, b(right) – discovered process model

Lets consider example shown in Fig. 3. The resource `resA` is a process resource and invokes resource `resB` during its execution. Next, two events with IDs `a` occur in two different resources `resA` and `resB`. If we omit resource information they are indistinguishable form each other. In order to use basic AA we need to flatten the log, to make sure ID of such events in the system are unique, we are concatenate resource URI and local ID — `A::a` and `B::a`. AA takes one log file with multiple traces (instances) of exactly one process as its input. In order to use it, first we need to gather distributed resource log files and concatenate them into one file for each process found by algorithm from Sec. 2.1. The first problem with concatenation, of independently generated log files, is the order in which concatenation is performed. Different approaches to deal with this problem have different impact on the results. It is because the AA only uses flat order of events in a log file to determine if two events are executed in parallel, in some order, or are independent on each other. In consequence, simple concatenation of resource log files will result in violation of causal dependency of events. In the considered example (Fig. 3a) adding `resB` log file at the and of `resA` file will result in incorrect dependency relation between event `A::e` and `B::a`. This will lead to incorrect conclusion that these events are not independent. In order to deal with this problem there we need to perform another preprocessing phase of the event log in order to identify the communicing resources and appropriately concatenate event logs of subinvoked resources. We consider only synchronous communication so if a resource does not execute multiple parallel threads, all invocation events must be followed by corresponding response events (`A::b` and `A::c`). If there are two parallel threads, then all events in the second thread must be parallel to both the invocation and the response handling event (`A::d`). In order to concatenate log files and respect ordering relation among events in both resources (context dependency between two events in different resource `A::b` and `B::a`), and in addition to respect local ordering of events, we use previously calculated sets of context chains in Eq. 5. For each chain, and for each resource instance occurring in context chain, we look for communication events invoking and handling response (*communication pair* $CP = (start, end, ctx_1, ctx_2)$). Each of such pairs consists of: *start* - starting event (invoking event), *end* - ending event (response handling event), ctx_A - context of invoking resource and ctx_B - context of invoked resource. Thanks to that, during pre-processing phase we put all events in the invoked resource event log file between the starting event

and the ending event. Additionally, not to disturb parallel relation of concurrent events there is a need to generate additional traces, not originally included in log file. Lets consider example in Fig. 3. We search for all *CP*s in the log. The only found *CP* is: `A::b`, `A::c`, `A::07C` and `B::AGF`. Because we are dealing with synchronous communication, we add all events of resource instance `B::AGF` between the events `A::b` and `A::c` of resource instance `A::07C`. The problem occurs with event `A::d`, which is parallel to communication events in `resA`. In order to respect the parallel relation, we need to generate new traces (the minimal set of them) that will render all events in log file of `resB` to be also parallel to event `A::d`. To do that, we need to generate new process traces (instances) with respect to the following condition:

$$\forall_{CP \in Log_A}((e \mid\mid start \wedge e \mid\mid end) \implies (\forall_{f \in Log_B}(f \mid\mid e)), \text{ where } Log_A \text{ is invoking resource log and } Log_B \text{ is invoked resource log, and } a \mid\mid b \Leftrightarrow a \succ_L b \wedge b \succ_L a \text{ , where } L \text{ is some log} \tag{11}$$

As a result, new traces are generated and we can execute the AA for each process occurring in the log. The discovered petri net is shown at Fig. 3b. In comparison to the original model in Fig. 3a, the dotted places and arcs are not present. It is because, there is no longer relation between events `b` and `c` at `resA` after log preprocessing. This is a side effect of adding `resB` log. In conclusion, presented example shows that AA is able to mine processes based on a preprocessed resource-oriented log. Some drawback of this approach is that during mining interaction between resources, some local dependencies are lost. In context of global PM this is not an issue, because from a global point of view the workflow is in fact transfered to invoked resource.

5 Conclusions and Future Work

We have shown how our approach may be used to extend the current methods of PM and Web Service Mining discussed in Sec 1 to make them applicable in ROS. We have discussed how RESTful log, including interaction events of ROS, can be obtained, and further used to discover the process model in such systems.

Presented considerations leads to several conclusions and feature challenges. First, framework to obtain context enriched log concerns only ROS implemented accordingly to JSR-311. In the case of other technologies more work may be required. In future we would like to address this problem. Another direction of research is to develop algorithms dedicated for ROS that do not need to preprocess event log. This may lead to more accurate process models by using all information available in the log, like hierarchy relation along resources and/or message semantics. Finally, current PM methods work only with global process. In our approach to reconstruct process related information we discover multiple process resources but later we execute process discovery algorithm for each of them separately. Our current work concerns developing methods for discovering processes models based on multiple process logs.

References

1. van der Aalst, W.M.P., et al.: Process mining: a two-step approach to balance between underfitting and overfitting. Software and Systems Modeling (2009)
2. van der Aalst, W.M.P., et al.: Process mining manifesto. In: Daniel, F., Barkaoui, K., Dustdar, S. (eds.) BPM Workshops 2011, Part I. LNBIP, vol. 99, pp. 169–194. Springer, Heidelberg (2012)
3. van der Aalst, W.M.P., Weijters, T., Maruster, L.: Workflow mining: Discovering process models from event logs. IEEE Trans. on Knowledge and Data Eng. (2004)
4. van der Aalst, W.: Service mining: Using process mining to discover, check, and improve service behavior. IEEE Transactions on Services Computing (2012)
5. van der Aalst, W.M.P.: Process Mining: Discovery, Conformance and Enhancement of Business Processes. Springer Publishing Company, Incorporated (2011)
6. van der Aalst, W., Verbeek, H.: Process Mining in Web Services: The WebSphere Case. IEEE Bulletin of the Tech. Committee on Data Engineering (2008)
7. Barros, A., et al.: Correlation patterns in service-oriented architectures. In: Proc. of the 10th Int. Conf. on Fundamental Approaches to Soft. Eng., pp. 245–259 (2007)
8. Buijs, J., et al.: Towards cross-organizational process mining in collections of process models and their executions. In: BPM Workshops (2011)
9. Dustdar, S., et al.: Web services interaction mining. Tech. Rep. (2004)
10. Dustdar, S., et al.: Discovering web service workflows using web services interaction mining. Int. J. of Business Process Integration and Management 1, 256–266 (2007)
11. Fielding, R.T.: Architectural Styles and the Design of Network-based Software Architectures. Ph.D. thesis, University of California, Irvine (2000)
12. Gaaloul, W., Bhiri, S., Godart, C.: Research challenges and opportunities in web services (2006)
13. Günther, C.W., van der Aalst, W.M.P.: Fuzzy mining: Adaptive process simplification based on multi-perspective metrics. In: Alonso, G., Dadam, P., Rosemann, M. (eds.) BPM 2007. LNCS, vol. 4714, pp. 328–343. Springer, Heidelberg (2007)
14. Hadley, M., Sandoz, P.: Jax-rs: Java api for restful web services (2008)
15. Ingvaldsen, J.E., Gulla, J.A.: 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)
16. Khan, A., Lodhi, A., Köppen, V., Kassem, G., Saake, G.: Applying process mining in soa environments. In: Dan, A., Gittler, F., Toumani, F. (eds.) ICSOC/ServiceWave 2009. LNCS, vol. 6275, pp. 293–302. Springer, Heidelberg (2010)
17. Kiczales, G., Hilsdale, E., Hugunin, J., Kersten, M., Palm, J., Griswold, W.: Getting started with aspectj. Communications of the ACM 44(10), 59–65 (2001)
18. Motahari-Nezhad, H.R., Saint-Paul, R., et al.: Event correlation for process discovery from web service interaction logs. The VLDB Journal 20(3), 417–444 (2011)
19. Mueller-Wickop, N., Schultz, M.: Erp event log preprocessing: Timestamps vs. accounting logic. In: vom Brocke, J., Hekkala, R., Ram, S., Rossi, M. (eds.) DESRIST 2013. LNCS, vol. 7939, pp. 105–119. Springer, Heidelberg (2013)
20. Pautasso, C., et al.: Restful web services vs. "big" web services: Making the right architectural decision. In: Proc. of the 17th Int. Conf. on WWW, pp. 805–814. ACM (2008)
21. Richardson, L., Ruby, S.: RESTful Web Services. O'Reilly Media (2007)

User–Defined Rules Made Simple with Functional Programming

Sava Mintchev

Baring Asset Management, 155 Bishopsgate, London EC2M 3XY, UK
sava.mintchev@barings.com

Abstract. To be successful, any new business information system must address the needs of business users, and have a short 'time–to–value'. Depending on the requirements, the appropriate tools and techniques would vary. Sometimes a good way to meet the needs of business users is by providing them with a domain–specific language (DSL) in which they can model their problems or seek solutions.

In this paper, we discusses our experience of an industrial project for the development of a corporate information system. A small DSL has been created using the Haskell functional language. The DSL has given business users the required degree of flexibility and control. The development was completed on time, and has confirmed Haskell's expressive power and the high performance of its compiled code. We also argue that Haskell is relevant to parallel Big Data processing, and to Decision Modelling applications.

Keywords: data analysis, integration, domain–specific language, business rules, decision modelling, functional programming, Haskell.

1 Introduction

Much of the potential value of Big Data is hidden in the insights which can be gleaned from it. To realise this value, IT specialists need to collaborate with domain experts, in order to devise, continuously apply and fine-tune the most appropriate data analysis methods. One of the main challenges of Big Data today is to make such multi-disciplinary collaboration as effective and productive as possible.

In a recent survey [17], business executives were asked why companies may be holding out on using Big Data. The top 3 answers were:

1. Need more education on how Big Data solves business problems (62% of respondents)
2. Need Big Data solutions to better address the needs of business users (53%)
3. Need better time–to–value for Big Data (47%)

In other words, executives want solutions to *specific business problems* and needs; and they want a quick return on investment in Big Data projects.

W. Abramowicz and A. Kokkinaki (Eds.): BIS 2014, LNBIP 176, pp. 229–240, 2014.

The same survey also addresses technology issues. When listing reasons for seeking commercial alternatives to Hadoop 2.0 (the popular open–source Big Data processing framework), most respondents cite "simplified data integration".

Business users are experts in their own field; they are not trained data analysts, computer programmers, statisticians, or experts in machine learning. Many business users would probably point to Excel as their tool of choice; but Excel on its own is not a tool for big data storage or analysis. Many other popular end-user tools and underlying technologies are also inappropriate ([6], [7]). The challenge for IT is not only to provide adequate technology, but also to help bridge the gap between business users and that technology.

In this paper we discuss the experience of addressing such challenges in a recent project for development of a data analysis system at Barings[1]. The business problem is summarised first, followed by an outline of the system design. Specific attention is given in subsequent sections to the approach to providing business user-definable rules for data analysis.

2 The Business Problem

The aim of the project was to deliver a new system for evaluation of the company's products (mutual investment funds) relative to each other, and within the wider universe of funds offered by other providers. The evaluation is based on the characteristics of funds, their historic performance, as well as market / sector trends.

Previously, such analysis had been carried out partially, manually, using external data providers' reporting and BI tools, and Excel (see Fig. 1). There were a number of problems:

- manual, inefficient and error prone processes
- inability to combine data from multiple sources to achieve deeper, multi-dimensional analysis
- lack of scalability - could be applied for a few funds only
- data and calculation maintenance headaches

In order to perform the required analysis fully, over 10^6 rows of data need to be processed. While such a volume is not truly "Big Data" by modern standards, it is sufficiently large to make analysis in Excel impractical.

A new, automated, scalable and flexible solution was needed, and a project was started. A traditional "waterfall" approach was applied to this project. In the initial feasibility stage, requirements were gathered by a business consultant working together with representatives of the relevant business areas: product development, investment management, sales, performance measurement. Then

[1] Baring Asset Management provides investment management services in developed and emerging markets to clients worldwide. The company operates from 11 countries, and has around 100 investment professionals, covering equity, bond and alternative asset classes. It is a subsidiary of MassMutual, a leading diversified financial services organisation.

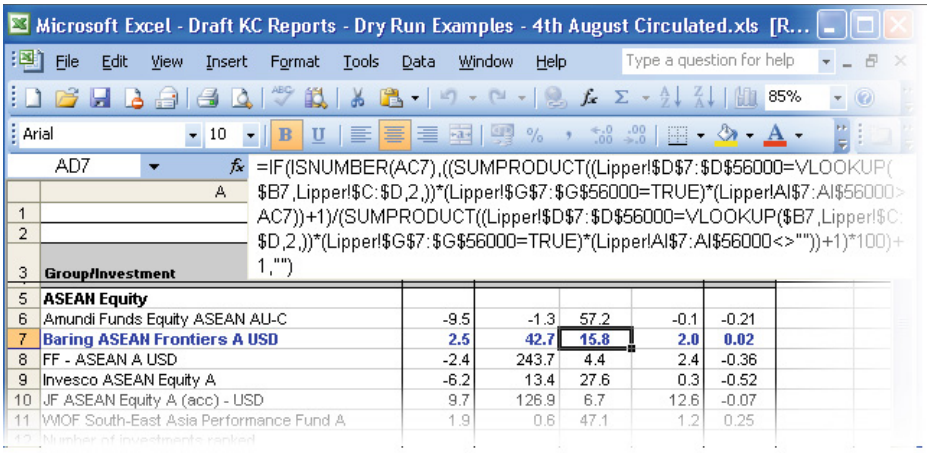


Fig. 1. Old pre-project Excel fragment

a search was conducted in the marketplace for existing solutions (software packages or external services) fitting the requirements. As no appropriate off-the-shelf solution was found, the project was scheduled for in-house design and development.

3 System Design

The new system has had to provide a range of functionality including data integration, calculation of analytics, report production, and user interface. It was decided at the outset that the system would be built of independent modules, all using a dedicated shared database. Each module could utilise a different technology. In this way we have been able to choose the most appropriate technology for each part, and utilise our existing investment in infrastructure, development tools and expertise. The modular system structure is illustrated in Fig 2.

The web-based **User Interface** controls the execution of all other system modules. The **ETL** (Extract–Transform–Load) module processes data from multiple sources: two external data providers (fund returns, flows, sales *etc.*), an internal performance system, and data spreadsheets. It transforms incoming data, resolves dependencies between different data sources, and populates the common data store, housed in a relational database management system (**RDBMS**). Both the User Interface and ETL modules have been created using the webMethods suite from Software AG, in line with our existing integration strategy [11].

The **Analytics Calculation** engine applies the methodology (as agreed by the business) for calculating statistical measures and analytics based on fund data: relative return, risk, momentum, track record, saleability, sales productivity, *etc.* This module has an Object–Relational mapping layer (Hybernate) [12], and is exposed as a webMethods service. The **Score Calculation** module is

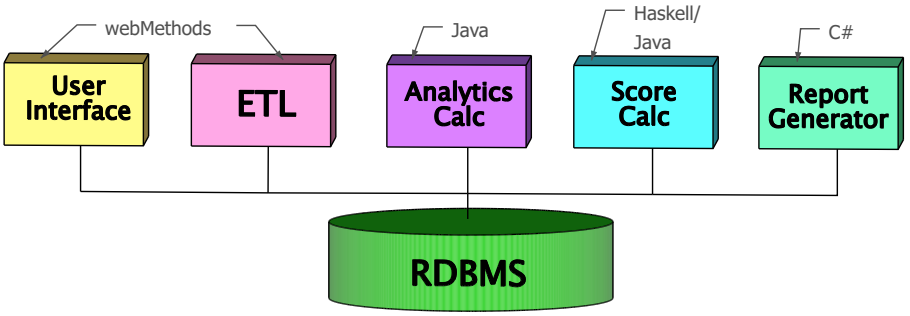


Fig. 2. System structure

discussed in the next section. The **Report Generator** creates a set of reports in strictly predefined Excel formats.

4 User-Definable Rules

An important requirement for the new system has been to enable business users to formulate and maintain *scoring rules*. The rules are used to formally rate each product – investment fund – against other funds within a given category. Each rule shows how to calculate a new value from underlying product characteristics, calculated analytics, and the output of other rules.

The following fragment shows a simplified representation of 3 such rules (`BAND`, `PERF_HL` and `PERF`) defined in terms of underlying product characteristics (`mstarRating`) and other rules defined elsewhere (`MD_HL` – *market demand*, `perfMDscoreSum` – *performance plus market demand score*, `KCGRANK1Y` – *key comp group rank over 1 year*, `QRANK1Y` – *quartile rank*, `REL_RETURN` – *relative return*):

```
BAND
  | PERF_HL  >= 2 && MD_HL >= 1 && perfMDscoreSum >= 3
              = 2
  | PERF_HL  >= 1 && MD_HL >= 0 && perfMDscoreSum >= 2
              = 1
  | otherwise = 0

PERF_HL
  | PERF >= 3.80    = 2
  | PERF >= 2.35    = 1
  | otherwise       = 0

PERF = mstarRating * 0.25 + KCGRANK1Y * 0.25 + QRANK1Y * 0.25 +
       REL_RETURN * 0.25
```

The first two rules represent step functions, while the third is a weighted average. The names (`BAND` *etc*) are cryptic, but come from the domain experts (business users) themselves.

To meet the business requirements, we have had to give end users the means to formulate their own scoring rules, and provide a rule evaluation module. The fragment of scoring rules can be seen as code in a functional programming language – Haskell; and in principle, a Haskell interpreter or compiler can be used for evaluating such rules.

From a different viewpoint, scoring rules can be seen as statements in a simple domain-specific language (DSL). An interpreter for such a language can easily be created in a lazy functional programming language like Haskell.

4.1 Representation of Scoring Rules

The sample rule fragment from Sect. 4 uses Haskell syntax, but this is not how our business analysts and users had envisaged their scoring rules to be formulated. In the Business Requirements document, different types of rule were presented in different tabular formats. So in the example from Sect. 4 , the `BAND` and `PERF_HL` rules were in a table called "conditional" rules, while the `PERF` rule was in another table called "scoring factor" rules.

A proposal for representing scoring rules in Haskell was put forward to Business consultants and users, offering them greater expressive power. It was declined, on the grounds that rules defined in fixed-format tables would result in "*the right balance between flexibility and change control*". In other words, business users would have the freedom to define new rules within the predetermined formats, and any changes to those formats would be made by developers in the IT department.

This leads us to consider how the sample rules from Sect. 4 can be represented in Haskell data types. The declarations can be quite straightforward[2]:

```
data ScoringRule = SR Id [ScoringRuleAlt]

data ScoringRuleAlt
    = ConditionalRule [Condition] Double [Note]
    | ScoringFactor [FactorWeight]
```

where `SR` is a *constructor*, and `Id` is a data type of rule identifiers. A rule has a list of alternatives (`ScoringRuleAlt`), corresponding to *guards* in Haskell syntax. The two alternative constructors (`ConditionalRule` and `ScoringFactor`) represent the two types of rule from Sect. 4.

The numeric values which rules are applied to, as well as the results of rule evaluation can be stored in a type of `ScoreValue`:

```
data ScoreValue = SV Id Double [Note]
```

[2] The code given here is simplified for clarity.

4.2 Scoring Rule Evaluation

We mentioned in Sect.4 two possible approaches to the evaluation of scoring rules:

1. Use a standard Haskell compiler or interpreter (*in which case scoring rules would be written in Haskell*)
2. Write an interpreter (*of a small language which scoring rules would be written in*)

Given that business users would not write their scoring rules in Haskell, the first approach loses its attraction: in order to use a standard compiler or interpreter, we would first have to read and pretty-print the scoring rules in Haskell syntax. We have decided against such an approach because:

a) In the context of our organisation, it is undesirable for a development tool (*e.g.* a Haskell compiler) to be used in a Production environment (*i.e.* beyond the development or test environments);
b) Using a standard compiler, it would be more difficult to provide execution trace and error messages which are clear and meaningful to a business user.

So we have decided to write an interpreter for scoring rules. This is quite easy to do in a lazy functional language:

```
calcScores :: [ScoringRule]->[ScoreValue]->[ScoreValue]
calcScores scoringRules scoreValues =
  let newScoreValues = map (calcScoringRule scoreValues') scoringRules
      scoreValues' = scoreValues ++ newScoreValues
  in newScoreValues
```

The `calcScores` function gets a list of rules (`scoringRules`), and a list of pre-calculated values. It returns a list of new score values, by evaluating every rule (`map (calcScoringRule...)`) in the context of all values (`scoreValues'`) – both pre-calculated values (`scoreValues`), and the results of rule evaluation (`newScoreValues`).

The function which evaluates a single rule has the following type signature:

```
calcScoringRule :: [ScoreValue] -> ScoringRule -> ScoreValue
calcScoringRule scoreValues scoringRule =
    SV (getScoringRuleId scoringRule) val notes
```

where `val` (definition is omitted for brevity) is the floating point number to which `scoringRule` evaluates; `notes` are associated with the applicable alternative of the rule; and `getScoringRuleId` is a de-constructor:

```
getScoringRuleId :: ScoringRule -> Id
getScoringRuleId (SR id ruleAlts) = id
```

4.3 The Joy of Laziness

An interpreter for a simple language can be written in many languages, and it is worthwhile considering what difference Haskell makes. Because scoring rules

can be given in any order, and one rule can refer to the results of other rules, an interpreter needs to evaluate rules in an appropriate sequence. One way of achieving this would be to apply (albeit simple) dependency analysis to the set of scoring rules.

The `calcScores` function from Sect.4.2 does not involve dependency analysis. The complete function declaration is perhaps the simplest way of stating what the rule evaluator is; it can be seen as a formal specification.

Moreover, the lazy evaluation semantics of Haskell ensures that `calcScores` is also an *executable* specification. In the `calcScores` function, using the result (`newScoreValues`) in its declaration only makes sense because of lazy evaluation. To see that, consider the declaration of `calcScoringRule`: it returns a `ScoreValue` with an identifier which comes straight from the rule being evaluated. Therefore the list of new score values (the result of `calcScores`) can safely be unwound, and the identifier of each element can be inspected.

4.4 Polymorphism and Higher–Order Functions

The `calcScores` function from Sect. 4.2 is meant for evaluating a specific set of rules - those described by the `ScoringRule` data type. Using the features of Haskell, it is straightforward to turn that function into a "generic" evaluator (`calcScoresGen`) for other sets of rules:

```
calcScoresGen :: ([a] -> b -> a) -> [b] -> [a] -> [a]
calcScoresGen calcScoringRuleX scoringRules scoreValues =
  let newScoreValues = map (calcScoringRuleX scoreValues') scoringRules
      scoreValues' = scoreValues ++ newScoreValues
  in newScoreValues
```

The only difference compared to `calcScores` is that we have added the function `calcScoringRuleX` of type `([a] -> b -> a)` as an argument to `calcScoresGen`. Here "`a`" can be any type of "score value", and "`b`" is an arbitrary type of "scoring rule".

Of course, in our example the definition of `calcScoresGen` is extremely simple; but one can imagine a more sophisticated function which, by virtue of polymorphism and higher–order functions, is still applicable to different data types, representing different domain-specific languages.

Hopefully the reader can see that the approach of using a DSL interpreted in a functional language has rather bigger potential relevance than the simple example of scoring rules might suggest.

5 Implementation

A simple rule evaluator along the lines of `calcScores` from Sect. 4.2 could be written in Haskell in hours; in our case, it was done incrementally, on and off within a week. The source code totalled under 500 lines.

Another few days were spent on a database access layer, and on exposing the rule evaluator as a service. In total, the development and unit test of the score

calculator took 8 man–days effort over a period of 3 weeks (interleaved with other projects). This represents just under 10% of the development effort for the whole project.

5.1 Program Compilation

The scoring rule evaluator was developed in Haskell, and compiled using the Glasgow Haskell Compiler (GHC) [10]. We have also produced a *naive*, unoptimised translation into Java which preserves the lazy evaluation semantics of the original Haskell code.

Table 1 shows the lines of code in the original programs (.hs), the Java translation (.java), and GHC-generated C code (.hc).

Table 1. Score Calculator – Lines of code

Module	**Description**	**LOC**		
		.hs	**.java**	**.hc**
ScoreDefs	*Data type declarations, access methods*	115	710	6,500
ScoreParse	*Parsing functions*	125	1,100	3,050
ScorePrint	*Pretty-printing*	50	390	1,300
ScoreCalc	*Interpreter for rules*	180	1,500	4,730
	Total	470	3,700	15,580

5.2 Integration with Other Modules

Haskell provides interoperability with other languages via the Foreign Function Interface (FFI) and is implemented in GHC for C/C++. There are additional tools developed by members of the Haskell community to facilitate interfacing to other languages, including Java. There are also packages (HDBC and others) for connecting to database servers. In the simplest case, a process started by an OS command can use standard Haskell IO for data communication via files or pipes. The Network library in Haskell can also be used.

The scoring rule evaluator is a stand–alone module in an application built in other languages. As part of the project, we would not have been able to evaluate different approaches; in this case, the simplest route of standard Haskell IO would suffice. However, given that we have translated the Haskell program into Java code, we could quite easily write additional Java code which links to the translated program. We have used this approach to call functions in the (translated) Haskell program from other Java code in two ways:

- Use database access in Java when calling translated Haskell functions;
- Expose a translated Haskell function as a Java service in webMethods

5.3 Experimental Results

The execution time of the scoring rule evaluator is shown in Table 2 for different size test data sets.

Table 2. Score Calculator – Run time

Number of scores	**GHC code run time** (*sec*)
10^3	0.02
10^4	0.1
10^5	1
10^6	10

Tests were run on a 2.4 GHz Intel Core 2 Duo processor machine under Mac OS. Haskell was compiled with "ghc -O".

6 Discussion

The work was completed successfully. Haskell's strict static type checking helped ensure that there was not much that could go wrong during testing. The maturity of Haskell tools – in particular of the Glasgow Haskell Compiler, GHC – has been apparent and reassuring, and the performance of compiled code has been impressive. The GHC compiler has been continuously developed and improved for over 20 years, and incorporates vast amounts of research in functional programming language implementation.

Since project completion in 2012, the system has been in active use. In our experience of other projects, subsequent enhancement requests are quite common soon after the go-live date, and further development is often needed. This has not been the case for this project. The flexibility which the new system provides to business users (to define their own rules using a simple DSL) has made further development unnecessary to date.

The claimed "simplicity" of the approach is twofold. First, business users and analysts have been able to design their own rules, without having to worry about implementing them (e.g. in Excel, or in a BI tool). Business users therefore envisaged a simple solution which followed their preferred data analysis methodology, and gave them the desired flexibility. Second, from an IT development perspective, the choice of a functional language – Haskell – made it relatively easy to create a DSL for user–defined rules. In Sect 4 we have tried to show how the expressive power of Haskell made the implementation simple.

A drawback of this approach is the reliance on specific programming language skills; there are not nearly as many Haskell programmers as there are Java or SQL programmers. Another drawback comes from the need to integrate a module written in Haskell with the rest of the system which uses different technologies.

6.1 Related Work

A review of modern business intelligence technology is presented in [2]. The system discussed in this paper has elements of a typical BI architecture, like Extract–Transform–Load (ETL) tools, Relational DBMS, Analytics calculation engine, Front-end and reporting applications. In this project there was not a requirement for online analytical operations (filtering, aggregation, drill-down, pivoting) or advanced visualisation, as described in [14]. On the other hand we had very specific reporting requirements, fixed in terms of content and layout. It was a conscious design decision not to employ an OLAP server, thereby foregoing the potential benefits of powerful BI tools. However we do have OLAP capability elsewhere in our IT architecture [11], *e.g.* for financial management information.

The use of DSLs for facilitating the collaboration between between domain experts and IT developers has been well established [5]. The design and implementation of DSLs embedded in Haskell has also been an area of productive research and development [1]. Our implementation is not 'embedded'; it is comparatively simple and straightforward, but has nonetheless helped to meet a real business requirement.

In the realm of business rules and decision support, there has been recent progress towards bridging the gap between Business and IT [15]. The *Decision Model and Notation* (DMN) specification has been submitted to the OMG, and has just been adopted and published in draft. The specification's main goal is to define an industry standard notation for decision management and business rules which is understandable by business users, analysts, and IT developers. The new specification is related to the Business Process Model and Notation (BPMN) OMG standard, in that BPMN decision tasks can be modelled with DMN. The DMN specification concerns a special type of business rules, and as such is more concise and application–focused than the Semantics of Business Vocabulary and Business Rules (SBVR) specification. DMN covers both modelling and execution aspects. We think that it would be appropriate to consider the implementation of DMN's *Friendly Enough Expression Language* (FEEL) – which is free of side effects – as a DSL in a functional language such as Haskell.

How relevant is a pure functional language like Haskell to the challenges of Big Data? It can be quite relevant, due to its support for parallel and concurrent programming [9], and the high performance of compiled code. With respect to concurrent programming, researchers have found that "*applications built with GHC enjoy solid multicore performance and can handle hundreds of thousands of concurrent network connections.*" [3]. But it is the parallel programming capability which makes Haskell particularly relevant to Big Data processing. In [8], the author reverse–engineers Google's MapReduce programming model [4], and creates a formal executable specification in Haskell. The specification is then refined to model parallel execution opportunities. The author also considers Google's domain–specific language Sawzall from a functional specification perspective.

It is said that writing MapReduce routines in languages like Java, Python or Ruby is rather more difficult than writing relational database queries in SQL

[16]. There are different approaches for making this task easier [2]. With its expressive power and support for parallel programming, Haskell could prove to be a good tool for MapReduce programming.

7 Conclusion and Future Work

In this paper we have discussed our experience in addressing some of the challenges associated with Big Data processing: data integration, and involvement of business users. We have applied a modular approach, employing different technologies and languages which are appropriate for the different stages in data processing. We have focused in particular on the use of a small domain-specific language (DSL) with an interpreter written in Haskell, a pure functional programming language.

The functional language development described here, while self-contained and completed successfully, is only a modest beginning for future work. The same approach can be adopted in other cases where there is a requirement for evaluating declarative rules. There are other possible applications of a similar rule engine in the context of the investment management industry, for example:

- Scoring of securities in equity research / quantitative analysis;
- Ensuring compliance with various regulatory rules and client–specific mandate restrictions;
- Calculation of client fees and rebates – the methods of fee calculation can vary from client to client;
- Master Data Management functionality – flexible rules for market data validation and "golden copy" construction

For wider industrial use, stability of the programming language and compilers is extremely important. Haskell is now a mature language with a stable specification and compliant implementations, most notably GHC. It is backed and developed by a wide community of researchers and industry practitioners [13]. Some of its features have been adopted by other languages, thereby becoming more familiar; and it is taught at an increasing number of universities. With its high–performance parallel implementation, Haskell can be a serious contender for effective Big Data processing.

Acknowledgments. The author is grateful to Stuart Holden, Stefan Holes, Stephen Schwartz, Mohanad Al-Bazi and other colleagues who have worked on this project.

References

1. Augustsson, L., Mansell, H., Sittampalam, G.: Paradise: a two-stage DSL embedded in Haskell. In: ACM Sigplan Notices, vol. 43, pp. 225–228. ACM (2008)
2. Chaudhuri, S., Dayal, U., Narasayya, V.: An overview of Business Intelligence technology. Communications of the ACM 54(8), 88–98 (2011)

3. Collins, G., Beardsley, D.: The Snap framework: A web toolkit for Haskell. IEEE Internet Computing 15(1), 84–87 (2011)
4. Dean, J., Ghemawat, S.: MapReduce: simplified data processing on large clusters. Communications of the ACM 51(1), 107–113 (2008)
5. Ghosh, D.: DSL for the Uninitiated. Communications of the ACM 54(7), 44–50 (2011)
6. Jacobs, A.: The pathologies of Big Data. Communications of the ACM 52(8), 36–44 (2009)
7. Labrinidis, A., Jagadish, H.V.: Challenges and opportunities with Big Data. Proceedings of the VLDB Endowment 5(12), 2032–2033 (2012)
8. Lämmel, R.: Google's mapreduce programming modelrevisited. Science of Computer Programming 70(1), 1–30 (2008)
9. Marlow, S.: Parallel and concurrent programming in Haskell. In: Zsók, V., Horváth, Z., Plasmeijer, R. (eds.) CEFP. LNCS, vol. 7241, pp. 339–401. Springer, Heidelberg (2012)
10. Marlow, S., Jones, S.P.: The Glasgow Haskell Compiler. In: Brown, A., Wilson, G. (eds.) The Architecture of Open Source Applications, vol. II (2012), http://www.aosabook.org
11. Mintchev, S.: Open it for business: Transforming information system infrastructure with a commercial BPM suite. In: Abramowicz, W. (ed.) BIS 2011. Lecture Notes in Business Information Processing, vol. 87, pp. 230–241. Springer, Heidelberg (2011)
12. O'Neil, E.J.: Object/Relational Mapping 2008: Hibernate and the Entity Data Model (edm). In: Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, pp. 1351–1356. ACM (2008)
13. O'Sullivan, B., Stewart, D.B., Goerzen, J.: Real World Haskell. O'Reilly Media (2009), http://book.realworldhaskell.org
14. Risi, M., Sessa, M., Tucci, M., Tortora, G.: CoDe modeling of graph composition for Data Warehouse report visualization. IEEE Transactions on Knowledge and Data Engineering 26(3), 563–576 (2014)
15. Taylor, J., Fish, A., Vanthienen, J., Vincent, P.: Emerging standards in decision modeling. In: Intelligent BPM Systems: Impact and Opportunity. BPM and Workflow Handbook Series, Future Strategies Inc., Lighthouse Pt (2013)
16. Teplow, D.: The database emperor has no clothes: Hadoops inherent advantages over RDBMS in the Big Data era. Business Intelligence Journal 18, 36–39 (2013), http://tdwi.org
17. uSamp. 2013 Big Data in Business Study. Study fielded by uSamp (United Sample Inc.), commissioned by 1010data (December 2013), http://info.1010data.com/Whitepaper-2013BigDataStudy.html

Risk Awareness in Open Source Component Selection

Mirko Morandini, Alberto Siena, and Angelo Susi

Fondazione Bruno Kessler
I-38123, Trento, Italy
{morandini,siena,susi}@fbk.eu

Abstract. Adopting Open Source Software (OSS) components offers many potential advantages – such as cost effectiveness and increased reputation – but also introduces a variety of new risks related to the intrinsic fluidity of the OSS development projects. In this paper, we present results of a systematic literature review on OSS adoption risks, which allows to relate them to available OSS measures. Relying on the results of the review, we also present a risk-aware selection process, which uses OSS measures to rank OSS project according to the adopter's criteria, improving the quality of the OSS component selection.

Keywords: Open Source Software, Software Component Integration, Risk Assessment.

1 Introduction

Complex business information systems often rely on several tenths or hundreds of software components to provide higher level business functionalities. Choosing the right components is a critical decision, as it could contribute to the success of the development project, or ratify its failure. Making the right decision requires to evaluate both the technical aspects of the components, such as functionality and quality, and the strategic aspects, including direct and indirect costs and the possible impact on high level objectives. In this component selection lies also the choice between Commercial Off-The-Shelf components (COTS) and Open Source Software (OSS) components.

At a superficial look OSS components are software components providing functionalities similar to COTS, but offered without a price. They have the potentiality to ease the achievement of an adopter's business objectives — such as maintenance cost reduction, software quality improvement or a possible increase of reputation. In fact, in 2013 an estimated 85% of all commercial software packages includes OSS (Gartner 2011). On the other hand, the intrinsic properties of OSS require substantial changes in the company-internal component evaluation processes. OSS projects are developed in open and distributed development communities, guided by heterogeneous personal and business objectives, and provided without quality of service agreements or formally commitments on the future roadmap. An uncontrolled adoption introduces risks of unplanned and

W. Abramowicz and A. Kokkinaki (Eds.): BIS 2014, LNBIP 176, pp. 241–252, 2014.

unbearable difficulties, costs, and efforts in the later phases of software development, in particular during long-term maintenance, contrasting the expected benefits of OSS. In 2011, inadequate risk management was identified among the top 5 mistakes to avoid when implementing OSS-based solutions, and failure rates in OSS projects were reported to be higher than 50% [9]. Understanding, managing and mitigating OSS adoption risks is therefore crucial to avoid potentially significant adverse impact on the business, in terms of time to market, customer satisfaction, revenue, and brand image.

In the present paper, we give a thorough view of risks in OSS component adoption and an approach to risk-aware decision making for component adoption. First, we detail the differences between OSS and commercial components and the resulting issues and opportunities. Second, aided by the results of a systematic literature study (SLR) we identify and organise the risks that a software development company needs to be aware of when integrating OSS components. For several of these risks, available indicators, measures and mitigation activities are presented. Finally, we present an enterprise level decision making process, which is tailored to assess risks exposure for managing OSS component adoption. It is centred around the concepts of OSS measures and indicators, and uses an ad-hoc implemented quantitative analysis algorithms.

The paper is structured as follows: Section 2 describes the peculiarities of OSS component adoption, risks, their measures and indicators. Section 3 sketches the proposed, tool-supported decision making process. Related work is presented in Section 4, while Section 5 discusses the achievements, highlights the knowledge gaps and outlines the ongoing challenges.

2 OSS Risks and Measures

We performed a systematic literature review (SLR), with the objective of getting a comprehensive view on existing risks in OSS vs. COTS components adoption, and to derive measures for our analysis. The SLR was conducted following the guidelines by Kitchenham [8], defining: 1. the purpose and intended goals of the review; 2. the details of the literature search; the search strategy includes: (a) definition of the publication channels to be searched (primary conferences, journals, other traceable sources of knowledge); (b) the definition of search terms and selection of the libraries to be searched; (c) the definition od selection criteria, which are used to determine which studies are included in, or excluded from, a systematic review; 3. relevance and quality assessment procedures (screening for exclusion), describing what will be the selection criteria to remove papers of insufficient quality or that are out of the domain, e.g. with checklists; the assessment is made, in this order, on title, abstract, introduction and the full text; 4. the data extraction strategy that defines how the information required will be obtained systematically; 5. the quantitative or qualitative analysis of the extracted data, in a way that it could be independently reproduced, and the writing of the detailed results of the review.

The review follows the general process defined by Kitchenham [8], searching for journal and conference articles in popular meta-libraries, with well-defined

search terms, is based mainly on the keywords *Open Source Software*, *Risk*, and *Adoption*[1]. Out of 332 retrieved papers, screened based on title, keywords, and abstract, 47 were selected for a detailed analysis. Besides giving a comprehensive overview on the state of the art in the domain of Open-Source-Software adoption, the analysis contributed to the realisation of glossaries of terms including relevant risks, measures, mitigation activities, analysis and validation techniques. The 47 selected papers can be classified in the topics OSS component adoption, OSS in general, off the shelf components and development, software risk analysis, software quality assurance and three SLRs. More than 50% of the papers validated their results on empirical data; most of them on surveys, analysis of available data, or experiments with subjects. In the following, we briefly explain the results relevant in the context of the present work.

2.1 Differences between COTS and OSS

We identify the main differences between commercial closed source and community-based OSS adoption, as reported in a number of observational and survey studies (e.g., [13,14,18,6]), as a starting point for focussing on the relevant aspects for an OSS adoption risk analysis process.

- The organisational context, which is missing a central player which is, in traditional component of the shelf development, the owner of the software, the main responsible for its quality and future development, and the contact for business agreements and support. Liability and trust are usually bounded to such actors.
- The development process usually applied in OSS communities, which emerges from the need of working in a distributed and heterogeneous environment where software is built bottom-up, not preceded by requirements specifications and architectural design phases.
- The motivation and objectives of the participating community members. They do not refer to a strictly organised hierarchy and there is often few incentive to accomplish tasks such as documentation, testing and certification.
- The data freely available in OSS, from the code and its metadata in repositories, to forums and internal communication, which may be a potential source of risk, but is also an important source of measures for quality and risk evaluation.

Software Evaluation Metrics. More than in COTS, where contracts, certifications, proper marketing, support assurance and a direct interaction with the vendor are used to build a trusted and to some level protected environment, in OSS adoption, many factors such as indirect costs, software quality, and the state of the community, and the roadmap, are not easily perceivable at the time

[1] Details on the design, references to the selected papers and the results of the systematic literature study can be found in a technical report available at `selab.fbk.eu/riscoss_ontology/riskSLR.html`

of software choice. Also, software quality metrics are mostly obtained by human reporting [22] and thus hardly available in OSS. Moreover, the uncoupling of companies, developers and software gives rise to issues with which traditional risk management processes, which base on maturity models such as CMMI, are not able to cope, as evidenced in various empirical studies [16].

Alternatively, liability and support can be commissioned to a certain degree to external companies (e.g. Red Hat for Linux). In this way, OSS components can go through the companies' well-established, more or less formally implemented COTS evaluation processes. This is not always possible and often not the favourable solution, both from the point of view of costs and additional risks. Several works propose maturity evaluation and quality management processes tailored to OSS, such as the OpenSource Maturity Model (OMM) developed in the QualiPSo project [5], which evaluates OSS maturity regarding code, process, license and evaluation quality and standards.

2.2 Risks in OSS

With the selection and subsequent integration of any software component into the own system, an organisation takes a range of opportunities and risks. Regarding the organisational context in which development and maintenance of an OSS takes place, issues such as a lack of ownership, unclear liability and responsibility and lack of professional support providers, need to be tackled (e.g. [4,6]). The applied development process and the community objectives lead to issues such as a lack of a roadmap, the response to customisation needs, the bug fixing time and the hidden costs that come from incomplete documentation, and the need for specialised training (e.g. [11,6]). Although often these risks are conceptually similar to those found in closed source and component of the shelf (COTS) development, they are characterised by addressing various particularities of open source software.

Building on the results of the SLR, we defined a comprehensive hierarchy of risk categories for adoption of OSS software components. To give a structured and concise overview, here we report the main risks identified, organising them on the bass of the different phases of the OSS component adoption.

1. **Component selection risks**
   (a) Selection effort ill-estimation
   (b) Risk of wrong component selection
      – Risk of not satisfying own functional requirements
      – Risk of not satisfying own quality requirements
      – Selection process risk
      – Uncertainty about OSS community organisation and governance
2. **Component integration risks**
   (a) Integration effort ill-estimation
   (b) Risk of component integration failure
   (c) Risk of deployment issues including complex multi-component arrangements and risk of incompatibility.
   (d) Risk of impact on final product quality

   - Risk regarding the impact of component integration to the functionality and quality of the final product
   - Risk regarding negative effects of additional interdependencies
   - Risk of impact on final product documentation and translation
3. **Component operation and maintenance risks**
   (a) Maintenance effort ill-estimation
   (b) Risk of component maintenance failure
      - Risk of failure for increased maintenance cost and effort
      - Lacks in own human capital preparation, training and skills
      - Risk of missing community support and productivity
      - Risk of update failure at new component release
      - Risk of not obtaining proper commercial support
   (c) Risk of having insufficient software quality and functionality
      - Risk of loosing the necessary component quality
      - Security risk: risk of hostile attack or accidental compromise, for obtaining, modifying or deleting sensitive data, caused by security vulnerabilities, architectural weakness, or human error.
      - Risk of reduced control over software evolution
4. **Legal risks**
   (a) Intellectual property risk
   (b) Risk of license issues infringement
   (c) Liability risk, risk of having to take responsibility for the integrated code.

2.3 OSS Measures

A main objective of our work is to reveal the factors of uncertainty in OSS, to provide companies with a comprehensive risk analysis. Risk awareness and monitoring are important pillars towards building trust in OSS. The better the openly available project data can be explored to this aim, the more the risk assessment result would be reliable. To quantify various qualities of the source code, data on bugs and bug fixing, on the community and its future roadmap, and on the adopting organisation, a variety of measures is proposed in the SLR papers. While the first three are exploring the data available in OSS repositories, the latter consider (partly subjective) non-repository measures.

- **Code metrics** (code complexity, API usage, change detection, software quality metrics, repository access measures), release delivery.
- **Bugtracker metrics** (bug number and correction time, mean time between software failure, etc.).
- **Community and Roadmap metrics** (developer and user community measures, project activity, ongoing efforts and roadmap, mail metrics, developer interactions, active power).
- **External measures: expert opinions, information on involved companies and available support**
- **Measures regarding the adopting company and its business values**

Various recent projects, such as *Flossmetrics* and the ongoing *OSSMeter* have a specific objective on OSS metrics, provide data repositories and comparison.

2.4 Risk Indicators

With a focus on the differences between OSS and closed source components identified in the SLR, we try to understand the causal relationships between risks and failures, risk metrics, and risk mitigation.

For evaluating and taking decisions based on risk levels, it would be advantageous to obtain estimates on risk event likelihood from objective, data-based sources. The rich set of accessible OSS data seems to be promising for finding reliable indicators able to estimate risk event likelihood, both for a risk-aware software component selection, and to monitor the software during its life-time, to take corrective mitigation actions, preventing failure.

Even if, some risks, especially higher level financial and market risks, will clearly be assessable only by the help of expert opinions [18], several works in the SLR tried to find statistically significant indicators for various risks to anticipate failure. The challenge herein is the missing information on failure of software adoption, which is often confidential or undocumented, or has multiple causes which cannot be isolated. Expert opinion, albeit being subjective, can however give an important contribution for creating and evaluating risk indicators.

Searching correlations between repository measures and software quality evaluated by experts, the Qualipso project [5] defined several quality indicators, which can be directly related to the corresponding risks. Results show e.g. that trustworthiness increases with lower method coupling, reliability is higher for software with a high number of commits and where most files do not need revisions, while performance decreases with the lack of cohesion between methods. [19] found good indicators for the risky changes: the number of lines being added, the bugginess of the files being changed, the number of linked bug reports and the developer experience. For reliability prediction, [10] achieved a high accuracy by combining module coupling and the number of use cases. Indicators are also found for API change risk, bug risk, quality and trustworthiness (e.g. [13]).

2.5 Risk Mitigation Activities

Few publications in literature speak about risk mitigation for OSS components. It is mentioned in form of general hints, such as developing the existing stock of human capital [11], doing peer reviews [4], following general COTS adoption decision and cost evaluation processes, evaluating the community and the similarity to previous projects [22], or to make managers aware of risks and opportunities [19]. However, no work could be found, which showed evidence for causal relationships between risk indicator values and the effect of mitigation.

3 A Measure-Driven Risk Evaluation Process

When a company ("the Adopter", from now on) has to decide about a software component, it generally has a large number of state-of-the art components at its disposal. A typical selection process (validated in our project by a collaboration with the industrial partners) can easily discard those that do not have the

desired technical qualities. For example, supposing that an Adopter wants to select a component to provide wiki functionalities into a final product, we have tested existing tools, also available online[2], and we have easily identified three wiki components that match some chosen requirements of implementation language (Java), project status (mature) and size (large); namely: (i) Mediawiki; (ii) Dokuwiki; and (iii) XWiki. However, our whole work is motivated by the problem that, *if at the time of the selection the three components appear to be equally appealing*, the different ways they are developed – different communities, different licensing, different funders and so on – *hide in each candidate component some risks that could come out at a later time.*

For example, in the case of the wikis we focus on two categories of risk First, if the *OSS community does not generate sufficient information*, the assistance and evolution cost may rise excessively, thus compromising the cost-effectiveness of the company, which reflects on its return on investment. Second, if the *OSS community is not active enough*, evolution may be limited, thus causing a failure of the innovation requirement; this may disrupt the company's reputation, thus preventing it from having future access to governmental acquisitions.

Assessing Risks. So the relevant question is: how can risk exposure be inferred from available OSS indicators? We answer relying on the knowledge available in the literature about risk and OSS qualities, as extracted from the SLR.

1. Gathering Measures. As said, measures (and other relevant information) are collected. from main data sources of the OSS project (such as jira and repositories, forums and mailing lists). Table 1 illustrates some of the measures that are potentially used to evaluate the three wiki OSS components mentioned above. The most available data are about the source code itself and the community message boards. Particularly interesting are the values, which are different from one component to another. For example, in our case, we found important differences in the number of commits and bug fixes (calculated per month); in the total number of blocking bugs still open at the time of our analysis (only available in one case); the activity of the mailing list (sensibly lower in one case); the toga number of lines of code; and the license.

Table 1. Some of the measures used to asses OSS components

	XWiki	MediaWiki	DokuWiki
Commits and bug fixes	240 p/m	550 p/m	50 p/m
Blocking bugs	stable ratio	unknown	unknown
Mailing list activeness	1000 p/m	1000 p/m	150 p/m
Size	700.000 LoC	1.000.000 LoC	<100.000 LoC
License	LGPL	GPL	GPL

[2] By means of `http://www.wikimatrix.org`, `http://www.ohloh.net` and each specific OSS project source code repository.

2. Complete and Customize. As shown, no exhaustive measures are available for every component. We must integrate measures with other information sources. *Expert judgement* successfully covers missing measures. Worth saying that often the decision maker plays the role of an expert. The ongoing work takes into account *learning* over time to reduce the need for experts. Due to lack of space we do not present here this part of the work.
3. Normalising Values. We rely on the measures acquired from OSS communities to perform a quantitative evaluation of OSS projects. However, measures, as well as user ratings and expert judgements, range in different intervals from each other. Moreover, measures are not necessarily available for every component, can differ by context and the tools used, and even having full availability of measures, as reported in the previous chapter, it is difficult and sometimes impossible to derive useful statistical correlations. To mitigate these limitations and ease the risk assessment, we (i) avoid to seek a strongly probabilistic approach, and opt for a simpler evidence-based one; and (ii) normalise measured values onto evidences.

An *evidence* is a numerical quantification of a certain truth value, expressed as a real number in the range [0..1], and represents the degree of confidence that the truth value holds, or does not hold. For example, with reference to "timeliness" (the capability of an OSS community to deliver releases according to its roadmap), we may having evidence 1 if the releases are always delivered on time, 0 if they are never delivered on time, and a value in-between if they are delivered on time only to a certain degree. An evidence can be derived from many sources. For example, from a statistical analysis, from an expert judgement, or even from a user assessment. This way, the concept of evidence can act as a bridge among the different sources. As a consequence, an evidence value should not be taken as informative *per se*, but has rather to be used in context, for comparison with other values. As the result of the normalisation phase, we end up with having a set of harmonised evidences.
4. Mapping Risks. The collected evidences are stored in an **evidence graph**, a data structure that allows us to support automated reasoning. An evidence graph is a directed graph comprised by nodes, arcs and connectivity functions. Each node of the graph represents a fragment of knowledge, and has a value of evidence associated to it. Initially, some nodes have evidence known, other do not. Arcs connect nodes with each other, are weighted, and their weight specify how the evidence is propagated from one node to the other. When more than one source node propagates a value to a target node, connectivity functions determine how the different propagations combine with each other (e.g., Average, Maximum and Minimum are connectivity functions). Such data structure can be evaluated by means of graph analysis algorithms such as Label propagation [15]. Similar to algorithms used for web indexing, such as PageRank, but much simpler, label propagation is a forward reasoning algorithm that starts from the knowledge about some known nodes of the graph, and has the objective of inferring knowledge about unknown nodes.

Figure 1 depicts an excerpt of an evidence graph built on the basis of the available measures and targeted risks. The graph is based on the risks identified

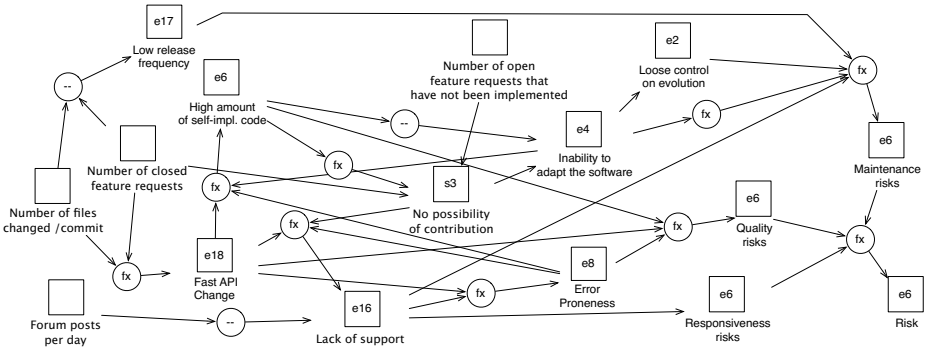


Fig. 1. Graphical representation of a part of the evidence graph used to evaluate risk exposure of the three OSS candidate components

in the SLR: every named risk event, cause or measure found in the SLR is represented as a node; every time in the SLR we found a correlation between risks, a link between a cause and a risk, or between a measure and a cause, an arc is added between the corresponding nodes. Notice that arc weights are currently the result of subjective estimation, and work is ongoing to double-check such estimation. The graph allows to relate available OSS measures, expert judgements and user input (if any) to possible risk events. Also, it allows to aggregate possible risk events to abstract risk classifiers. The graph contains input nodes, intermediate nodes and output nodes; however, there is no clear-cut distinction between intermediate and output nodes, since an intermediate node can also be informative for the user, acting this way as output node. Input nodes are generally those nodes that represent OSS measures, such as "Forum posts per day". Measure nodes trigger specific risk nodes, which correspond to possible risk events, such as "Fast API change". Specific risk nodes in turn trigger general risk categories. The depicted graph contains the "Maintenance risks", "Quality risks" and "Responsiveness risks" nodes. Finally, at the last aggregation level the graph contains a summary output node, the "Target risk" node. It contains the result of propagating values across the evidence graph. Notice that the topology of the graph may change if new knowledge is available.
5. Assessment. Once the evidence graph has been built, the candidate OSS components with respect to the highlighted dangers through the reasoning algorithm described above. The algorithm starts from the input nodes. Input nodes are generally those representing OSS normalised measures. For example, the node "Forum posts per day" is an input node. The more posts are made in a given OSS forum per day, the more the value of that node tends to 1.

Results. Example results of the analysis process are illustrated in Table 2. The results have been produced by means of a reasoning tool developed in Java. The tool takes as input a risk model, the functions and weights, and the input evidences, and produces as output the risk evidences. The table shows a hierarchy of risks. Each level of the hierarchy corresponds to an abstraction level. At the

root level there is the "Target risk" generic aggregator. It reports a number, which is the risk evidence resulting from the aggregation of the other risks. At the second level there are three more aggregate risks: "Maintenance risks", "Quality risks" and "Responsiveness risks". On the right part of the table, the risk values are reported for the three candidate OSS components. The values are intended to be used only for comparing the components with each other, so they do not have sense in terms of – for example – probabilities. This kind of result is intended to provide information about the evaluated OSS components, beyond those made available by simple analysis, and allowing to compare the components (and their development project) with respect to the information available online. Work is currently ongoing to evaluate the significance of available information and to improve it, in particular with respect to arc weights.

4 Related Work

Much work has been done in the literature about measuring OSS components, as reported in Section 2. In general, research on causal relations between failures, risks, measures and mitigation activities with statistical evidence, was performed mainly for security and business aspects, and a reliable and general correlations between risks and the predictive ability of measurements has been established only for few risks and software qualities. Several literature reviews were carried out in the domain of OSS adoption. [2] performed an SLR for identifying the different ways to adopt OSS, from the adoption of OSS-characteristic processes, to OSS use. Our efforts concentrate on component adoption on other software, here classified as integration. Similarly, [7], classified research on OSS with an SLR, identifying the business models with involvement of OSS. More relevant to our work, [20] analysed the challenges encountered in literature about OSS in development. The results are an important source of data, however, measures or mitigation activities were not considered. Since business information system development is largely model-driven, we focus on this class of risk assessment techniques to support risk-aware decision making. For example, CORAS [12] is a model-based approach based on UML for performing security risk analysis and assessment based on security analysis techniques, and is flexible to be

Table 2. Risk evidence for the three OSS components under analysis; notice that the reported values do not imply probabilities and do not have a meaning as standalone values: they are only intended to be used for comparison

	Xwiki	MediaWiki	DokuWiki
Target risk	**0.82**	**0.8**	**0.85**
Maintenance risks	0.52	0.5	0.7
Quality risks	0.38	0.35	0.4
Responsiveness risks	0.45	0.5	0.68
Lack of support			
Fast API change			
…			

supported by several possible techniques such as graph label propagation. The Goal-Risk framework [1] aims at capturing, analysing and assessing risk at an early stage of the requirements engineering process, using goal models. Similarly, the goal-oriented analysis methodology KAOS [21] deals with risk management by complementing goal analysis with obstacle analysis that consists in identifying and modelling the adverse conditions that may prevent a goal to be achieved. In [3], KAOS is extended to assess requirements risk. It has a formal representation of the whole goal model, relies on strictly measurable variables and takes into account partial or probabilistic values for goal satisfaction, but dues not integrate concrete measures and indicators. Risk assessment is performed to reduce the likelihood or the severity of each obstacle, and mitigation is performed by applying countermeasure patterns in the goal model. EKD [17] models organizational processes that can serve as a basis for identification risk management tasks. A central aspect is the development of enterprise knowledge models pertaining to the situations being examined.

5 Discussion and Conclusion

In this paper we presented a survey on the available literature in open source software (OSS) components adoption. The gathered risk knowledge, together with the measures that can be collected from OSS project websites, are used in a decision making process to build evidence graphs. Evidence graphs in turn are used to infer risk exposure of various OSS projects, and to rank them. So far, the approach has been successful in showing the correctness of some fundamental choices, but it is planned to apply it on the complete set of risks, with measures obtained from ongoing statistical analyses of OSS repositories. In particular, the approach strongly relies on the underlying knowledge about risks, measures and their correlations, to produce the necessary evidence graphs, which can not be proven to be fully correct. Although the user can incrementally update the graph according to his own knowledge, a need for deep modifications may limit its usefulness. At the current state of the work, the knowledge about OSS risks and measures is mainly based on the literature review, past projects that analysed this aspect and current practices and experiences in a limited set of organisations. More research is ongoing in the next two years within the EU FP7 project RISCOSS, to build a trustable base of knowledge and specialized tools.

Acknowledgement. This work is a result of the RISCOSS project, funded by the EC 7th Framework Programme FP7/2007-2013, agreement number 318249.

References

1. Asnar, Y., Giorgini, P., Mylopoulos, J.: Goal-driven risk assessment in requirements engineering. Requir. Eng. 16(2), 101–116 (2011)
2. Ayala, C.P., Cruzes, D., Hauge, Ø., Conradi, R.: Five facts on the adoption of open source software. IEEE Software 28(2), 95–99 (2011)
3. Cailliau, A., van Lamsweerde, A.: Assessing requirements-related risks through probabilistic goals and obstacles. Requir. Eng. 18(2), 129–146 (2013)

4. Ebert, C.: Open source drives innovation 24(3), 105–109 (2007)
5. Engineering- Ingegneria Informatica et al. Qualipso – quality platform for open source software. Project, IST-FP6-034763 (November 2006)
6. Hauge, Ø., Cruzes, D.S., Conradi, R., Velle, K.S., Skarpenes, T.A.: Risks and risk mitigation in open source software adoption: Bridging the gap between literature and practice. In: Ågerfalk, P., Boldyreff, C., González-Barahona, J.M., Madey, G.R., Noll, J. (eds.) OSS 2010. IFIP AICT, vol. 319, pp. 105–118. Springer, Heidelberg (2010)
7. Höst, M., Orucevic-Alagic, A.: A systematic review of research on open source software in commercial software product development. Information & Software Technology 53(6), 616–624 (2011)
8. Kitchenham, B., Charters, S.: Guidelines for performing systematic literature reviews in software engineering. Technical Report EBSE 2007-001, Keele University and Durham University Joint Report (2007)
9. Lee, S.-Y.T., Kim, H.-W., Gupta, S.: Measuring open source software success. Omega 37(2), 426–438 (2009)
10. Lee, W., Lee, J.K., Baik, J.: Software reliability prediction for open source software adoption systems based on early lifecycle measurements. In: COMPSAC, pp. 366–371 (2011)
11. Li, Y., Tan, C.H., Teo, H.H.: Firm-specificity and organizational learning-related scale on investment in internal human capital for open source software adoption. In: Proceedings of SIGMIS CPR, pp. 22–29. ACM, New York (2008)
12. Lund, M.S., Solhaug, B., Stølen, K.: Model-Driven Risk Analysis - The CORAS Approach. Springer (2011)
13. Morasca, S., Taibi, D., Tosi, D.: Towards certifying the testing process of open-source software: New challenges or old methodologies? In: Proceedings of FLOSS 2009, pp. 25–30. IEEE, Washington DC (2009)
14. Morgan, L., Finnegan, P.: Open innovation in secondary software firms: an exploration of managers' perceptions of open source software. SIGMIS Database 41(1), 76–95 (2010)
15. Nilsson, N.J.: Problem-solving Methods in Artificial Intelligence. McGraw-Hill, New York (1971)
16. Petrinja, E., Sillitti, A., Succi, G.: Adoption of oss development practices by the software industry: A survey. In: Hissam, S.A., Russo, B., de Mendonça Neto, M.G., Kon, F. (eds.) OSS 2011. IFIP AICT, vol. 365, pp. 233–243. Springer, Heidelberg (2011)
17. Rolland, C., Nurcan, S., Grosz, G.: Enterprise knowledge development: the process view. Information & Management 36(3), 165–184 (1999)
18. Rudzki, J., Kiviluoma, K., Poikonen, T., Hammouda, I.: Evaluating quality of open source components for reuse-intensive commercial solutions. In: SEAA 2009, pp. 11–19 (2009)
19. Shihab, E., Hassan, A.E., Adams, B., Jiang, Z.M.: An industrial study on the risk of software changes. In: Proc. of 20th Int. Symposium on Foundations of Software Engineering, FSE 2012, New York, NY, USA, pp. 62:1–62:11 (2012)
20. Stol, K.-J., Babar, M.A.: Challenges in using open source software in product development: a review of the literature. In: Proceedings of FLOSS 2010, pp. 17–22. ACM Press, New York (2010)
21. van Lamsweerde, A., Letier, E.: Handling obstacles in goal-oriented requirements engineering. IEEE Trans. Software Eng. 26(10), 978–1005 (2000)
22. Wahyudin, D., Min Tjoa, A.: Event-based monitoring of open source software projects. In: ARES 2007, pp. 1108–1115 (2007)

Continuous Quality Improvement in Logistics Service Provisioning

Martin Roth[1], Stefan Mutke[1], Axel Klarmann[1], Bogdan Franczyk[1,2], and André Ludwig[1]

[1] Information Systems Institute, University of Leipzig, Grimmaische Straße 12, 04109 Leipzig, Germany
{roth,mutke,klarmann,franczyk,ludwig}@wifa.uni-leipzig.de
[2] Institute for Business Informatics, University of Economics, Komandorska 118/120, 53-345 Wrocław, Poland
{bogdan.franczyk}@ue.wroc.pl

Abstract. Driven by rising competitive constraints companies began to outsource at least parts of their internal activities to specialized external logistics providers in terms of contracts. Thereby, new business models like the fourth party logistics evolved, which acts like coordinator of the emerging logistics networks. The main task of the provider is the planning, monitoring and measurement of the different networks of its customers. Based on a method to integrate process modeling and their simulation, complex event processing to gather real-time information about process executions and service profiles to measure subsequent service providers' quality – a closed loop approach for improving the quality of service provisioning is presented.

Keywords: quality management, logistics networks, simulation, complex event processing, service profiles.

1 Introduction

Nowadays, companies are faced with increasing competitive pressure, unpredictable market changes and dynamically changing regulations and business partners. These challenges must be handled in an adequate manner to fulfill processes faster, better, more flexible and in accordance to the requirements of the customer. Therefore, companies outsource their internal activities to external providers in order to reduce costs, increase profits or to focus on its core business. This evolution also occurs in the logistics service sector [1].

In recent decades, the logistics sector transformed enormously. While planning and monitoring of logistics services was formerly a task performed by one company, these activities are today realized by so called value-added logistics or fourth party logistics service providers (4PL) [2]. The outsourced logistics functions encompass basic services like transportation, handling and storing of goods but also value-added services like packaging, finishing or clearing goods. As a result, the planning and monitoring is very complex, whereby these tasks cannot be performed by a single

W. Abramowicz and A. Kokkinaki (Eds.): BIS 2014, LNBIP 176, pp. 253–264, 2014.

company, but by a network of specialized logistics service providers (LSP). Owing to the specific requirements of each company (quantity of demanded services or integration level) and due to each industry-specific requirement (tracing issues) every requested logistics service (LS) is individual in scope and quality. Within a network of affiliated logistics service providers a 4PL selects matching LSPs to the needed services and integrates them to meet the customer's requirements. Hence, a 4PL is defined as an independent, singularly accountable, non-asset based integrator of a client's supply and demand chains [3] by integrating upstream (e.g. suppliers) and downstream (e.g. distributors) actors of the supply chain [2]. A 4PL is the main contact person, coordinator of the involved logistics providers, and have the responsibility for the overall process and its quality of service.

To determine the quality of service, the 4PL has to monitor and to measure the performance of each participating partner. Fig. 1 illustrates a 4PL's service lifecycle which consists of the phases: analysis, design, implementation, operation and retirement [4].

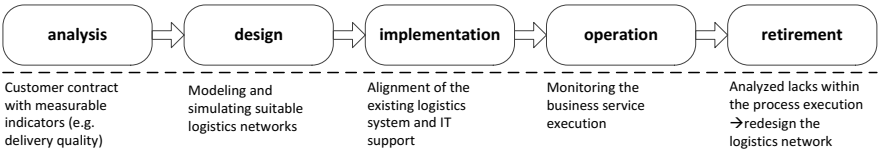


Fig. 1. 4PL service lifecycle

This contribution is organized as followed. First we introduce a closed control loop for continuous quality improvement based on the work of Deming including plan, do, check and act steps (Section 2). In Section 3, methods for implementing the continuous improvement cycle are outlined. Afterwards, it is shown how those approaches are integrated to form a continuous improvement cycle (Section 4). In Section 5, a scenario is used to illustrate the interaction between the presented approaches. The paper ends with a conclusion.

2 Continuous Improvement and the Approach for 4PL

The aim of this contribution is to describe a closed loop approach for the improvement of the service provisioning quality. Following the work of Deming the continuous improvement of processes is reached by using a four step sequence of activities [5]. Those activities are plan, do, check and act and are executed in a cycle, so that the result will be improved in every iteration (see Fig. 2).

The "plan" step orients towards improving the service provisioning by searching for deficiencies or potential optimizations, using e.g. process modeling and simulation. During the "do" step the changes planned in the previous step are implemented. To verify the validity of the implementation the execution of the changes has to be monitored and data has to be collected. The measurements aggregated during the "do" step are controlled in the "check" step. The results of the

execution are compared to the expected outcomes and baselines, which is often supported by visualization. If the results differ significant from the planned ones, corrective action has to be taken, which are chosen in the "act" step. If no corrective action has to be taken, the next iteration could be used to improve the quality further.

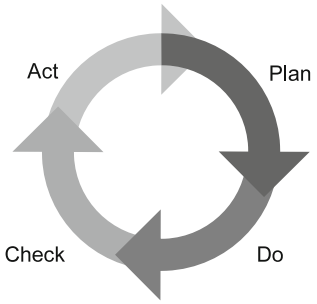


Fig. 2. PDCA Cycle

To apply this continuous improvement process in the context of a 4PL the sequence has to be integrated in the operational activities of the 4PL, which are focused on the construction and management of a service network. Therefore, based on the requirements, which are defined by the customer, the 4PL plans a process and selects the appropriate service providers. The selection of the provider is based on its capabilities and past performance. The planned process has to be annotated with the corresponding performance requirements and the feasibility has to be verified, e.g. using simulation with the existing service provider base (see analysis and design in Fig. 1). To perform the planned logistics process, the existing logistics information systems of the participating partners must be integrated (see implementation in Fig. 1). While the process is executed the performance of the whole process and the service instances should be measured (see operation in Fig. 1). The effectiveness of the execution and quality of the outcome is checked against the requirements of the customer and if an adaptation is necessary the process model is modified in an "act" step accordingly (see retirement in Fig. 1), which is followed by another cycle in the plan, do, check and act sequence.

3 Methods for Implementing the Continuous Improvement Cycle

To allow the implementation of the cycle described in the previous chapter, methods for planning, measuring and controlling are required. The following methods are already used to support the 4PL service life cycle (see [4], [6]), but a holistic approach is still missing. Therefore, we use the PDCA Cycle to show, how those methods can be used to improve the quality of logistics service provisioning continuously.

3.1 Simulation Approach for an Integrated Planning Process

Simulation approaches are widely used in logistics in order to plan logistics systems. Ingalls discusses the benefits of simulation as a method to study the behavior of logistics networks [7]. Additionally, advantages and disadvantages are presented for analyzing supply chains with the use of simulation. A concrete simulation approach is not provided. In [8] a commonly applicable simulation framework for modeling supply chains is presented. Instead of [7] they focus on a more technical perspective as they show an overview of event-discrete simulation environments in terms of domains of applicability, types of libraries, input-output functionalities, animation functionalities, etc. Cimino et al. also show how and when to use certain programming languages as a viable alternative for such environments. A modeling approach and a simulation model for supporting supply chain management are presented by Longo and Mirabelli in [9]. In addition, they provide a decision making tool for supply chain management and, therefore, develop a discrete event simulation tool for a supply chain simulation. All these approaches are relevant for developing an integrated planning and simulation approach. However, all these approaches satisfy the 4PL's specific requirements [6] only partially. The development of simulation models based on process models is insufficiently considered.

The planning of value-added logistics services is performed using several different models (e.g. process model, service profile, and simulation model). A rough plan, including each sub-service and their temporal dependencies, is represented by a process model. Based on this, dynamic aspects of logistics systems can by analyzed using simulation. The main task of simulation in logistics is to study the behavior of complex logistics services (e.g. lead times, transport volumes and capacities) to ensure that customer needs can be met. Thus, it is possible to analyze the flow of the goods through the logistics system with regard to the capacity to identify bottlenecks early on. In [6] specific goals and requirements concerning a 4PL´s simulation approach are discussed closer and the need for an approach to transform process to simulation models is highlighted. The most important requirement is the integration of the simulation approach in the entire planning process. The added value of the approach is, thus, the automatic transformation of existing process models to simulation models as described in the following.

A process model, e.g. Business Process Model and Notation (BPMN) or Event-driven Process Chain (EPC), is simulation independent, i.e. the model does not contain any information regarding to the dynamic aspects, such as arrival times, processing times or capacities. The process model is transferred into a transformation model and enriched with information required to run a simulation. However, the transformation model is platform independent and therefore cannot be executed in a specific simulation tool. The specific simulation models (e.g. Enterprise Dynamics[1] (ED), Arena[2]) are generated from the transformation model. The structure of the transformation model is described in more detail in [10]. Fig. 3 illustrates this approach.

[1] http://www.incontrolsim.com
[2] http://www.arenasimulation.com

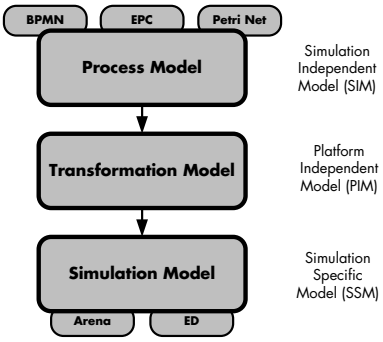


Fig. 3. Transformation procedure – From process to simulation

The acquisition of correct and robust information is an important success factor in the implementation of simulation projects. An initial approach for information acquisition in an integrated simulation approach for a 4PL was described in [11]. In this paper it will be shown, how service profiles are used to improve the quality of the planning and thus the provision of logistics services in networks.

3.2 Performance Measurement Using Complex Event Processing

In this section we outline how on-going contracts can be monitored in real-time by using concepts and methods of complex event processing (CEP). The acquired information is used in different contexts regarding to the 4PL service lifecycle and the included tasks. The consideration is, thereby, on an abstract level without going into detail of a particular tool or technique.

CEP is a relatively young discipline, whereby the most important concepts and tools already exist. Technical problems like scalability are mostly solved, but functional issues are not treated sufficiently [12]. The overall approach and the use within the logistics area, especially for the 4PL business model, are not covered in current research. Yao et al. analyze the use of CEP in hospitals using RFID [13]. They introduce a framework and provide a possible solution to improve patient safety. Some parts of the approach can be partly adopted, but the framework is too abstract and not suitable for the presented application area. Wang et al. introduced a framework for bridging the gap between physical and real world with the use of RFID and CEP [14]. This framework provides comprehensive support of RFID applications, but only covers a small part of the overall approach outlined in this paper. In [15], Zang and Fan describe how event processing can fit in enterprise information systems. They argue that with the evolvement of software architecture into SOA and the adoption of RFID, event processing can be an important player in enterprise information systems.

CEP is defined as a set of tools and techniques for analyzing and controlling the complex series of interrelated events. Thereby, events are processed as they happen, thus, continuously and in a timely manner [16]. An event is the central aspect of CEP and is defined as “anything that happens, or is contemplated as happening (change of

state)" [16], e.g. a RFID-enabled good is recognized by a RFID-reader. If an event summarizes, represents or denotes a set of other events, it is also called complex event, e.g. a RFID-enabled good left the issuing area [16]. In this paper it is assumed that CEP is already used to monitor an instantiated logistics network [4]. The next paragraph exemplifies this and emphasizes the adequacy of applying CEP in the area of a 4PL.

The outsourced service between the 4PL and the customer as well as between the 4PL and the LSPs is secured on a contractual level. A contract records the agreed upon obligations and responsibilities of all contractual parties in terms of business process conditions [17]. These conditions are often expressed as goals which must be achieved by each party. The goals can be extracted from the customers' requirements or from legal regulations and are known as SLOs, which define measurable indicators like delivery quality, delivery reliability or delivery flexibility. The contract describes the target state of each LS realized by the participants of the network and acts like a pattern. As soon as the process execution is started, the 4PL has to ensure the fulfillment of the defined SLOs. To achieve this, internal (e.g. good left the issuing area) and external (e.g. traffic jam) information regarding to the good will be pushed to the 4PL. By doing this the 4PL can ensure that possible penalties (e.g. delayed or damaged good) will be handed out to the "faulty" participant of the network. If it is not traceable which member of the network is the flaw, a logistics network would not be robust and sustainable over a longer period. In so doing, CEP allows to forecast, whether an instantiated process will meet the SLOs in the future or not [4]. For this purpose external cloud services like traffic, weather or timetable systems are integrated and processed. A simple example is: If the service execution reaches a certain point (e.g. GPS coordinates of a transport), CEP automatically checks if e.g. the departure time of the aircraft planned to use changed and if the overall contract can be fulfilled. To improve this forecasting capability, the information gathered from already completed service executions are analyzed for similarities to the ongoing contracts. In the case of a delayed departure the 4PL has enough time to take measures to compensate the delay (e.g. re-plan the service execution). This incoming information, which describes the actual state, is compared to the SLOs that depict the target state. This comparison takes place within the CEP engine. All information will be processed to evaluate the process execution of every logistics network partner and build up service profiles. The service profiles include key performance indicators, which benchmark the LSP and their services execution. In contrast to current approaches, this evaluation takes place during the process run-time and not at the expiration of a contract.

CEP is a powerful approach to process data, transform the process-accompanying data to information and link them to business processes. By doing so, CEP is a suitable technique to provide actual information at a desired granularity level in a timely manner. Hence, CEP is a suitable approach to measure the performance of logistics networks and to support the other lifecycle phases with latest data constituted by service profiles. This leads to a better database, whereby the 4PL has not to rely on experience or outdated information. The available information includes current performance profile built up from internal and external information and can be combined as desired. To evaluate the performance of each LSP, this information has to be transformed to service profiles, using indicator systems.

3.3 Service Profiles for Quality Control of Logistics Networks

Performance measurements, which are aggregated to indicators, carry information about business situations and are quantifying those situations as facts using a certain specificity [18]. The indicators are used to lead business decision towards business goals in an economical sense. [19] criticizes the making of decisions based on only one indicator, because it is not able to capture the facts in a closed relation and is not able to support coordination. Using loose indicators, without taking their respective relations into consideration, is also not efficient in reaching the economic goals. With the use of an indicator system, which incorporates the relations between the indicators, it is able to clarify inconsistencies as also dependencies in planning, control and monitoring [20]. Following [21], indicators carry certain roles. Those roles involve the function to inform about business facts and lead decisions, the comparing function (benchmarking), to compare different business units or businesses, the function of controlling the deviation of the current is-situation from the planned should-situation and the management, in the form of self-management or external-management [22]. A different approach on the differentiation of the indicators takes [23], which distinguishes between diagnostic, a more long-range view and analysis on the business situations, and an interactive, more real-time access to the description of the business situation, indicator systems. The levels of the objective function may be differentiated accordingly in an operational view, with its near real-time information concerning the whole business, and the strategic view, which focuses on a more far-sighted controlling of the reachability of strategic business goals. A further differentiation is possible by using the different kinds of relations between indicator systems. A calculation system allows the aggregation of one or more key performance indicators (KPIs), based on the formulas that integrate other lower-level indicators, which lead the business decisions. A service profile, in this respect, captures the quality in which a service provider performs its service over a certain time-span. This service profile is deduced from the agreement between contractor and service user. This service level agreement (SLA) guarantees a certain quality. The service level agreement incorporates some of the indicators to allow the attainment of standards with a specified deviation, in terms of service level objectives (SLO) [24]. The service profile is based on the aggregation of certain measurements, which are acquired during different runs of processes, which requires the indicator system to be a calculation system.

LogiBEST and SCOR are calculation systems, because they provide the necessary formulas to aggregate indicators to some key indicators, from the operations level to the strategic level, but are limited to two key indicators. The LogiBEST approach is especially valuable, because it integrates the variations of the performance, which is particularly important in the light of lean production systems. LogiBEST encompass standard degression and average deviation into the indicator system. The aggregation of this different deviation is not part of the indicator system, but is necessary for a business wide indicator system of a networked business. Concerning the requirement to have a consistent and logistics focused indicator system, the LogiBEST approach is the most useful one. Its only missing aspect is the aggregation of indicators and the respective aggregation of the variations. All studied indicator systems are lacking in the network internal view, with indicators like cooperativeness and trust, for which

the systems have to be extended. To allow a systematic handling of the service profiles of the networked businesses, it is necessary to model the constraints and relations of the indicator system to allow storage, filtering and display. Based on the approach shown in [25], which was developed with the SCOR indicator system in mind, we were able to design an appropriate model for the LogiBEST system, adapted to the mentioned requirements (see Fig. 4).

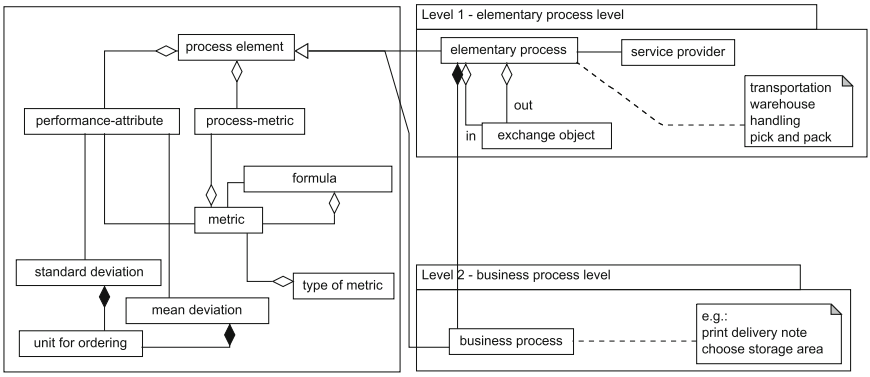


Fig. 4. Draft of the indicator systems

It removes those parts of the system, which are not essential for the 4PL viewpoint and extends the model with variations introduced in the LogiBEST system and perspectives described in the balanced score card approach. Furthermore, it is required to extend the LogiBEST system to a calculating system, which means defining the formula for aggregation of the indicators and its variations. Using the ideas of [26], which shows the aggregation of four types of indicators, it is possible to aggregate the indicators vertically, from operational to strategic, and horizontally, along the logistics chain. The indicators are, therefore, assigned to the types “additive” and “multiplicative”. The indicator “delivery reliability”, defined as the percentage of deliveries in time and quantity, is in this perspective a “multiplicative” indicator and the total “delivery reliability” of the logistics chain is calculated as the product of the reliabilities of its service components.

4 Integrating the Approaches to Form the Continuous Improvement Cycle

The methods described in the previous chapter are not sufficient to allow for a continuous improvement of the service provisioning if considered alone. Combined, they form the basis for a business information system, which allows for the improvement using the four step cycle of Deming [5]. Fig. 5 shows the integrated approach.

During the “plan” step the simulation approach of chapter 3.1 is used to verify the dynamic aspect of the process model and to determine the required performance of the respective service providers with regard to the requirements of the customer for a certain logistics network. Therefore, the process model is transformed into the

respective simulation model using the approach described above. The validity and feasibility of the process is verified using the service profiles of the existing service providers described in chapter 3.3. Therefore, the simulation and optimization of the process is based on a sound knowledge of the service providers. Based on a successful simulation, the SLOs for each participating partner are defined and contractually secured. These SLOs describe the target state of a service execution.

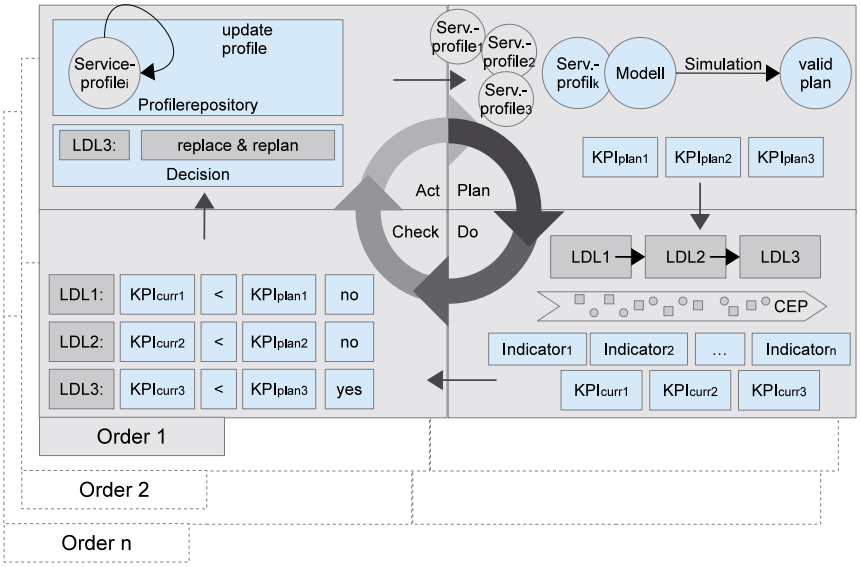


Fig. 5. Conclusion on the approach

During the execution of the process ("do" step), CEP allows the acquisition of real-time information concerning the performance of single process-steps as also of the process as a whole (chapter 3.2). The "check" step is implemented using the CEP approach again to compare the performance of each service provider against its SLO defined in the planning step. Using this information and an updated service profile after one execution the decision to act upon the deficiencies or potential optimization could be made, which leads to another improvement cycle. As a result, LSPs can be substituted, their service profile updated and the network may be rescheduled. This updated profile is afterwards used as a direct input for further orders. Thus, new orders can be planned with an up-to-date and reliable database. Therefore, the quality of all other networks managed by the 4PL is kept high.

5 Scenario

In this section the interaction of the presented approaches are illustrated with the use of the following scenario (see Fig. 6). The 4PL has the order to transport goods from Yunlin to Berlin. Based on the contractually secured overall reliability (90%) the 4PL plans and models the overall logistics service. Thereby, suitable LSP, which can

collaboratively fulfill the customer's logistics service, are selected. The 4PL simulates the modeled logistics service regarding to dynamic aspects like cycle time. If the results of the simulation meet the defined requirements, the 4PL ensures the execution of the services with each participating LSP by using SLOs ("plan" phase).

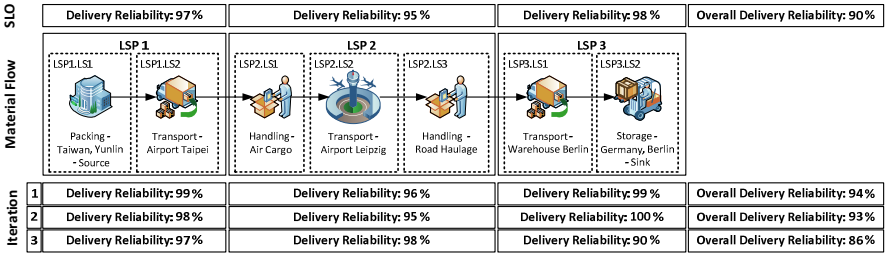


Fig. 6. Scenario with defined SLO

In this example, three LSPs are needed to perform the customer's order. As soon as the service is executed ("do" phase), data is generated and gathered using CEP. With the use of automatic identification technologies (AIDC), the volume of data increases enormously. Thereby, companies are only interested in information with high values. Hence, messages like "packet1 was read at Reader123 at 10:00 p.m." don´t have high information content and thus are useless for companies. These messages are generated as long as the RFID transponder remains in the RFID reading field. Irrelevant or redundant messages are filtered by CEP mechanisms. The increase of the information value is made by aggregation mechanisms, to transform technical to business-relevant events. By combining the technical events with other data sources, the message can be aggregated to "packet1 for Mr. X left the warehouse at 10:00 p.m. with a delivery delay of 30 minutes". This comparison between the target and the actual performance is done within the "check" phase. With the help of indicator systems, the gathered information is transformed and structured to service profiles. Overall, the service is carried out three times in this example. In the first iteration, the LSPs achieve delivery reliability of 99% (LSP1), 96 % (LSP2) and 99 % (LSP3). This results in an overall delivery reliability of 94%. Thus, in the first iteration, the customer's requirement is fulfilled. In the second iteration, the overall reliability decreases by 1%, whereby the contract between the customer and 4PL is not violated. In the third iteration, the overall reliability of supply is below 90% and thus the contract between the customer and 4PL is violated. Therefore, the 4PL is able to react on those circumstances and can prevent such breaches of contract in this or in other iterations ("act" phase). Moreover, the service profile of each LSP is updated after each service execution, which makes it possible to plan a more robust logistics networks for other clients.

6 Conclusion

In this contribution an integrated approach for continuous improvement in the context of 4PL-providing is presented. Therefore, the common PDCA cycle is adapted regarding to the characteristics of the 4PL service lifecycle. To present a holistic

approach, different methods for each phase of the continuous improvement cycle (PDCA) are introduced. To plan logistics networks, process models are transformed to simulation models. This forms the integrated planning approach. To achieve a robust and reliable simulation, the used data has to describe the latest performance of the involved partners and their service execution. Therefore, on-going contracts are monitored and compared to the defined conditions (SLOs) by using CEP. The presented holistic approach allows identifying positive and negative deviations in real-time, which are the base for measuring the performance of the involved partners. Thereby, it is possible to plan new logistics networks with realistic data of on-going contracts. To use this data for continuous improvement, a transformation based on an indicator system is needed. Due to the requirements of the 4PL business model, available systems must be extended, e.g. considering the interdependencies within logistics networks. Regarding to the continuous improvement of processes, an efficient combination of the presented artifacts arise, which form the basis for an improvement cycle in the light of Deming. Fig. 5 illustrates the interaction of the artifacts and the cross-order use of service profiles within the respective logistics networks. A scenario is used to illustrate the benefit of the overall methodology.

Acknowledgments. The work presented in this paper was funded by the German Federal Ministry of Education and Research under the project LSEM (BMBF03IPT504X) and LogiLeit (BMBF 03IPT504A).

References

1. Lockwood-Lee, J., Forey, G., Lockwood, J.: Globalization, Communication and the Workplace: Talking Across the World. Continuum International Publishing Group (2010)
2. Schmitt, A., Weber, P.D.J.: 4PL-ProvidingTM als strategische Option für Kontraktlogistikdienstleister: Eine konzeptionell-empirische Betrachtung. Deutscher Universitätsverlag (2006)
3. Win, A.: The value a 4PL provider can contribute to an organisation. International Journal of Physical Distribution & Logistics Management 38, 674–684 (2008)
4. Roth, M., Donath, S.: Applying Complex Event Processing towards Monitoring of Multi-party Contracts and Services for Logistics – A Discussion. In: Daniel, F., Barkaoui, K., Dustdar, S. (eds.) BPM Workshops 2011, Part I. LNBIP, vol. 99, pp. 458–463. Springer, Heidelberg (2012)
5. Deming, W.E.: Out of the crisis. Center for Advanced Engineering Study 6. Massachusetts Institute of Technology, Cambridge (1986)
6. Mutke, S., Klinkmüller, C., Ludwig, A., Franczyk, B.: Towards an Integrated Simulation Approach for Planning Logistics Service Systems. In: Daniel, F., Barkaoui, K., Dustdar, S. (eds.) BPM Workshops 2011, Part I. LNBIP, vol. 99, pp. 306–317. Springer, Heidelberg (2012)
7. Ingalls, R.G.: The value of simulation in modeling supply chains. In: Proceedings of the 30th Conference on Winter Simulation, pp. 1371–1376. IEEE Computer Society Press (1998)
8. Cimino, A., Longo, F., Mirabelli, G.: A general simulation framework for supply chain modeling: state of the art and case study. International Journal of Computer Science Issues 7, 1–9 (2010)

9. Longo, F., Mirabelli, G.: An advanced supply chain management tool based on modeling and simulation. Comput. Ind. Eng. 54, 570–588 (2008)
10. Mutke, S., Augenstein, C., Ludwig, A.: Model-based integrated planning for logistics service contracts. In: 17th IEEE International Enterprise Distributed Object Computing Conference, pp. 219–228. IEEE Computer Society (2013)
11. Mutke, S., Roth, M., Ludwig, A., Franczyk, B.: Towards Real-Time Data Acquisition for Simulation of Logistics Service Systems. In: Pacino, D., Voß, S., Jensen, R.M. (eds.) ICCL 2013. LNCS, vol. 8197, pp. 242–256. Springer, Heidelberg (2013)
12. Bruns, R., Dunkel, J.: Event-driven Architecture: Softwarearchitektur für ereignisgesteuerte Geschäftsprozesse. Springer, Heidelberg (2010)
13. Yao, W., Chu, C.H., Li, Z.: Leveraging complex event processing for smart hospitals using RFID. J. Netw. Comput. Appl. 34, 799–810 (2011)
14. Wang, F.-S., Liu, S., Liu, P., Bai, Y.: Bridging physical and virtual worlds: Complex event processing for RFID data streams. In: Ioannidis, Y., et al. (eds.) EDBT 2006. LNCS, vol. 3896, pp. 588–607. Springer, Heidelberg (2006)
15. Zang, C., Fan, Y.: Complex event processing in enterprise information systems based on RFID. Enterprise Information Systems 1, 3–23 (2007)
16. Luckham, D., Schulte, R.: EPTS Event Processing Glossary v2.0. Event Processing Technical Society (2011)
17. Weigand, H., Xu, L.: Contracts in E-Commerce. In: Meersman, R., Aberer, K., Dillon, T. (eds.) Semantic Issues in E-Commerce Systems, vol. 111, pp. 3–17. Springer (2003)
18. Reichmann, T.: Controlling mit Kennzahlen und Managementberichten: Grundlagen einer systemgestützten Controlling-Konzeption. Vahlen, München (2001)
19. Wolf, J.: Kennzahlensysteme als betriebliche Führungsinstrumente. Verlag Moderne Industrie, München (1977)
20. Horváth, P.: Controlling. Verlag Franz Vahlen, München (2003)
21. Weber, J.: Einführung in das Controlling. Schäffer-Poeschel, Stuttgart (1993)
22. Gladen, W.: Performance Measurement: Controlling mit Kennzahlen. Gabler-Verlag, Wiesbaden (2011)
23. Reinecke, S.: Marketingkennzahlensysteme: Notwendigkeit, Gütekriterien und Konstruktionsprinzipien. In: Reinecke, S., Tomczak, T., Geis, G. (eds.) Handbuch Marketingcontrolling. Marketing als Treiber von Wachstum und Erfolg, pp. 690–715. Wirtschaftsverlag Ueberreueter, Frankfurt/Wien (2001)
24. Sturm, R., Morris, W., Jander, M.: Foundations of Service Level Management. Pearson Sams, Michigan (2000)
25. Stein, A.: Erweiterung des Supply Chain Operations Referenzmodells. PhD Thesis. Universität Münster, Wirtschaftswissenschaftliche Fakultät (2010)
26. Jaeger, M.C., Rojec-Goldmann, G., Muhl, G.: Qos aggregation for web service composition using workflow patterns. In: Enterprise Distributed Object Computing Conference, pp. 149–159. IEEE Computer Society (2004)

Author Index